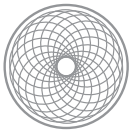


Pacemaker

Cluster from Scratch - Apache

Creating an Active/Passive and Active/Active Cluster on Fedora 11

by Andrew Beekhof

pacemaker@clusterlabs.org

Last updated on Thursday, October 22, 2009 for Pacemaker 1.0.5

Table of Contents

Read-Me-First

The Scope of this Document

The purpose of this document is to provide a start-to-finish guide to building an example active/passive cluster with Pacemaker and show how it can be converted to an active/active one.

The example cluster will use:

- Fedora 11 as the host operating system
- OpenAIS to provide messaging and membership services,
- Pacemaker to perform resource management,
- DRBD as a cost-effective alternative to shared storage,
- OCFS2 as the cluster filesystem (in active/active mode)
- The crm shell for displaying the configuration and making changes

Given the graphical nature of the Fedora install process, a number of screenshots are included. However the guide is primarily composed of commands, the reasons for executing them and their expected outputs.

Formatting conventions used in this document

command to be executed

```
[root@test1 ~]# command to be executed
Sample output
Part of the output to look for in particular
More output
```

If a command is listed and not followed by any output, then it should be assumed that the command does not produce any.

What Is Pacemaker?

Pacemaker is a cluster resource manager. It achieves maximum availability for your cluster services (aka. resources) by detecting and recovering from node and resource-level failures by making use of the messaging and membership capabilities provided by your preferred cluster infrastructure (either OpenAIS or Heartbeat).

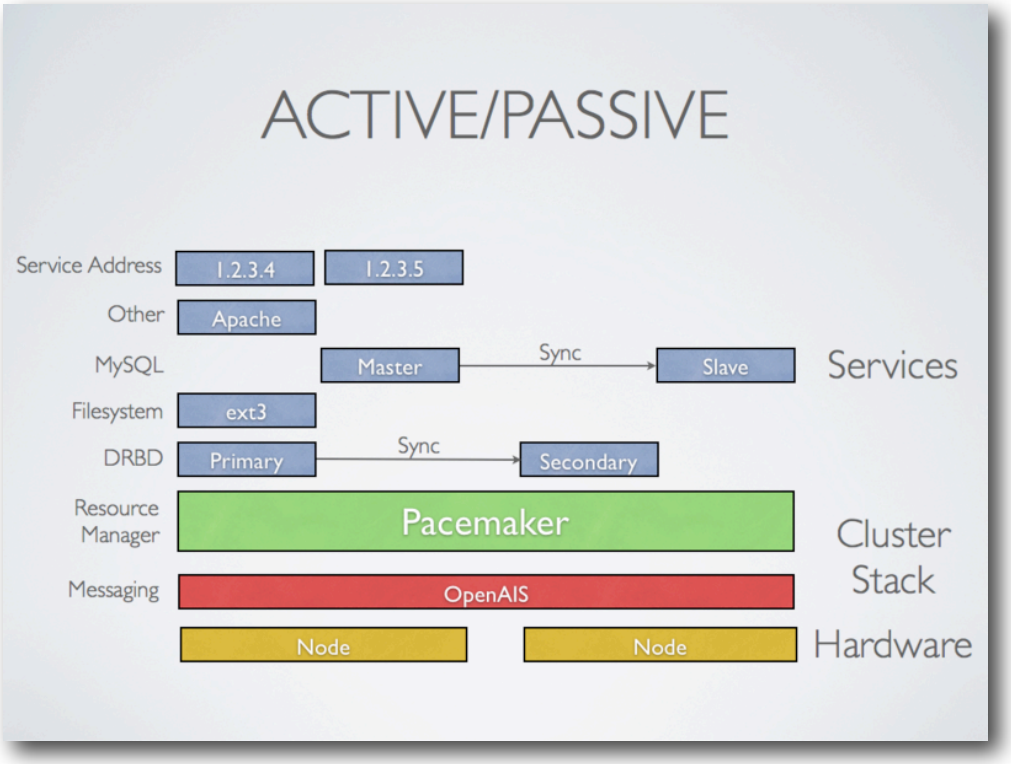
Pacemaker's key features include:

- Detection and recovery of node and service-level failures
- Storage agnostic, no requirement for shared storage
- Resource agnostic, anything that can be scripted can be clustered
- Support for both large and small clusters
- Optionally ensure data integrity with STONITH
- Ability to specify cluster-wide service ordering, colocation and anti-colocation
- Support for services which need to be active on multiple nodes
- Support for services with multiple modes (eg. master/slave, primary/secondary)
- Unified, scriptable, cluster shell

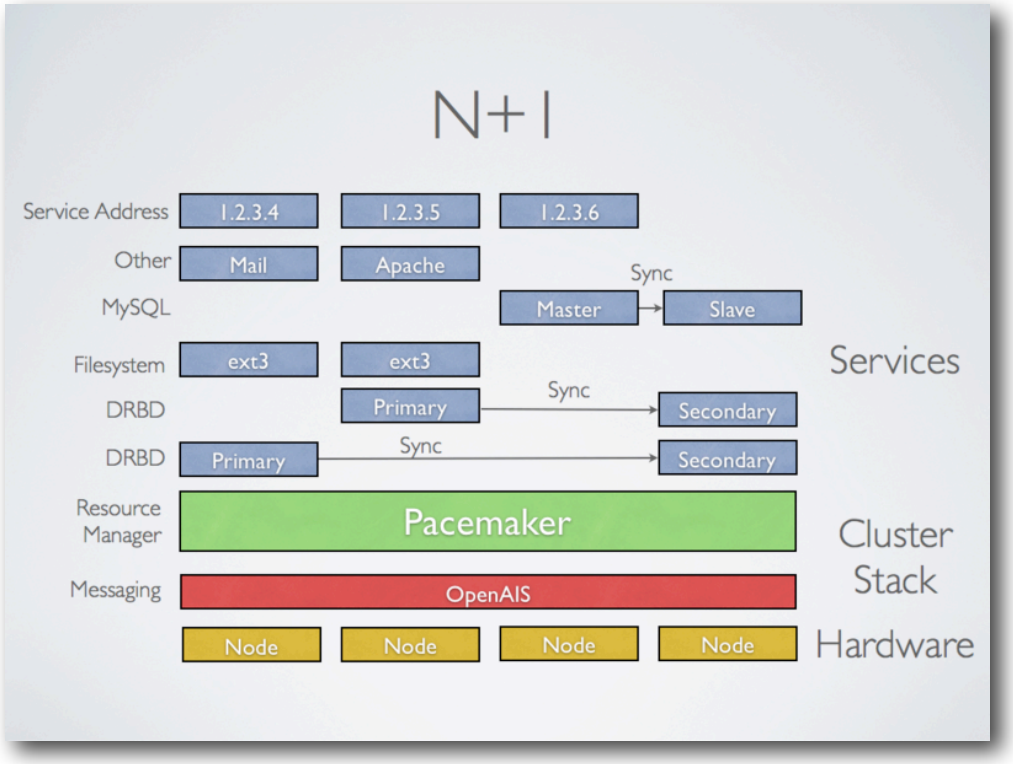
Types of Pacemaker Clusters

Pacemaker makes no assumptions about your environment, this allows it to support practically any redundancy configuration including Active/Active, Active/Passive, N+1, N+M, N-to-1 and N-to-N.

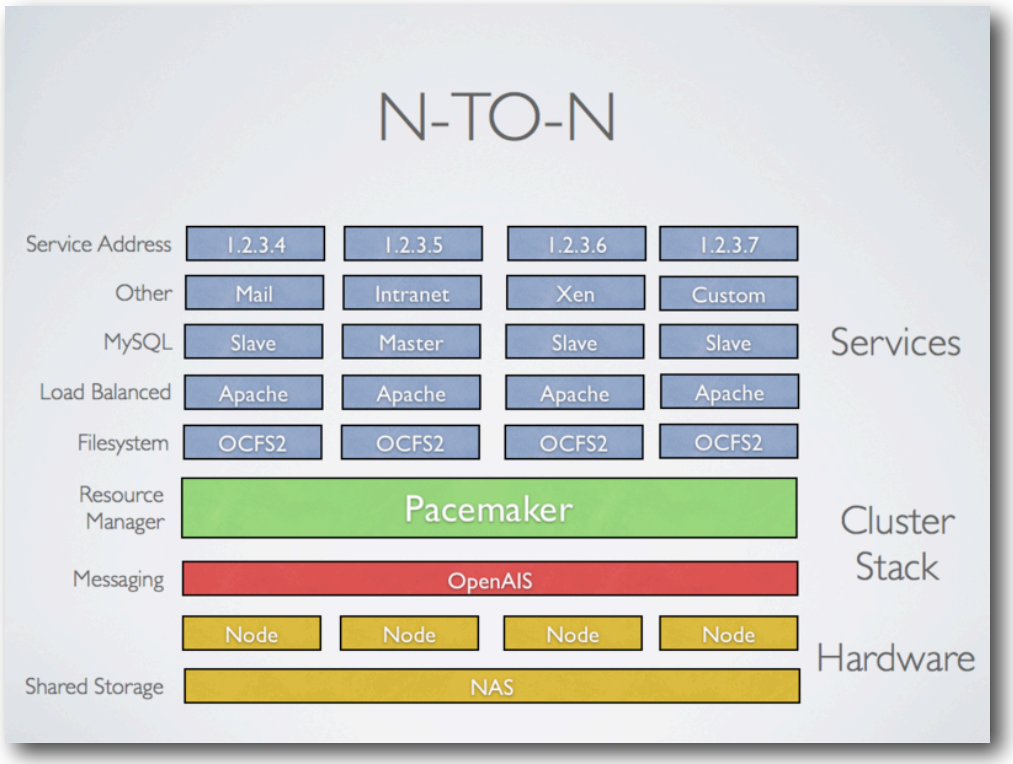
Two-node Active/Passive clusters using Pacemaker and DRBD are a cost-effective solution for many High Availability situations.



By supporting many nodes, Pacemaker can dramatically reduce hardware costs by allowing several active/passive clusters to be combined and share a common backup node



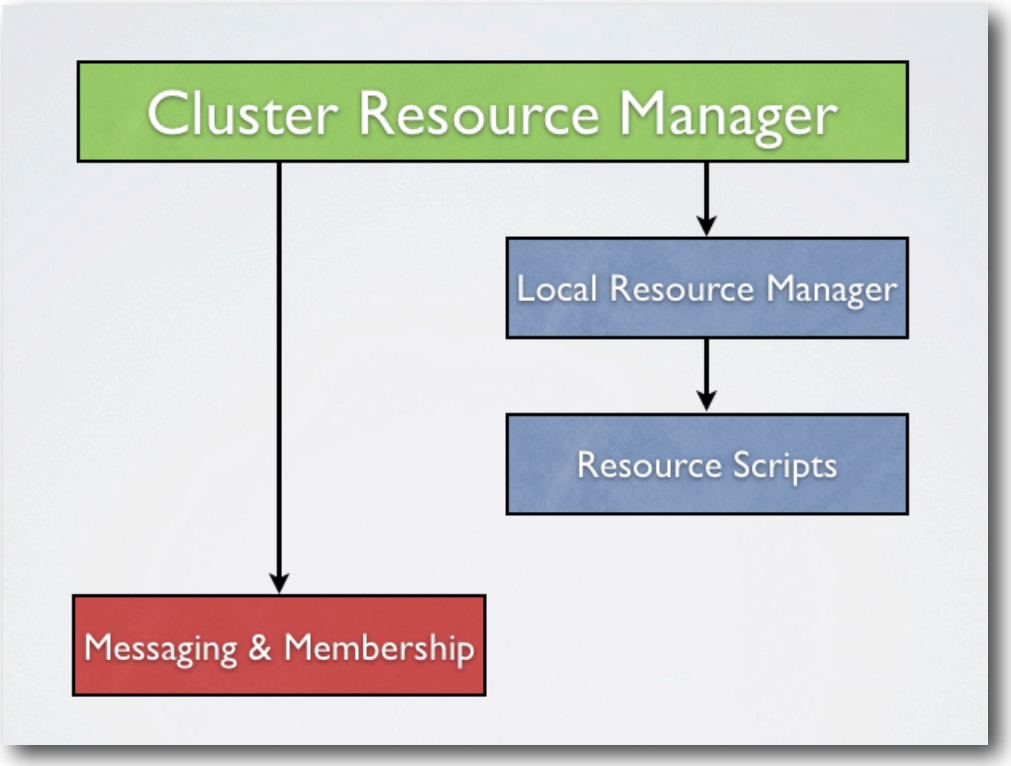
When shared storage is available, every node can potentially be used for failover. Pacemaker can even run multiple copies of services to spread out the workload.



Pacemaker Architecture

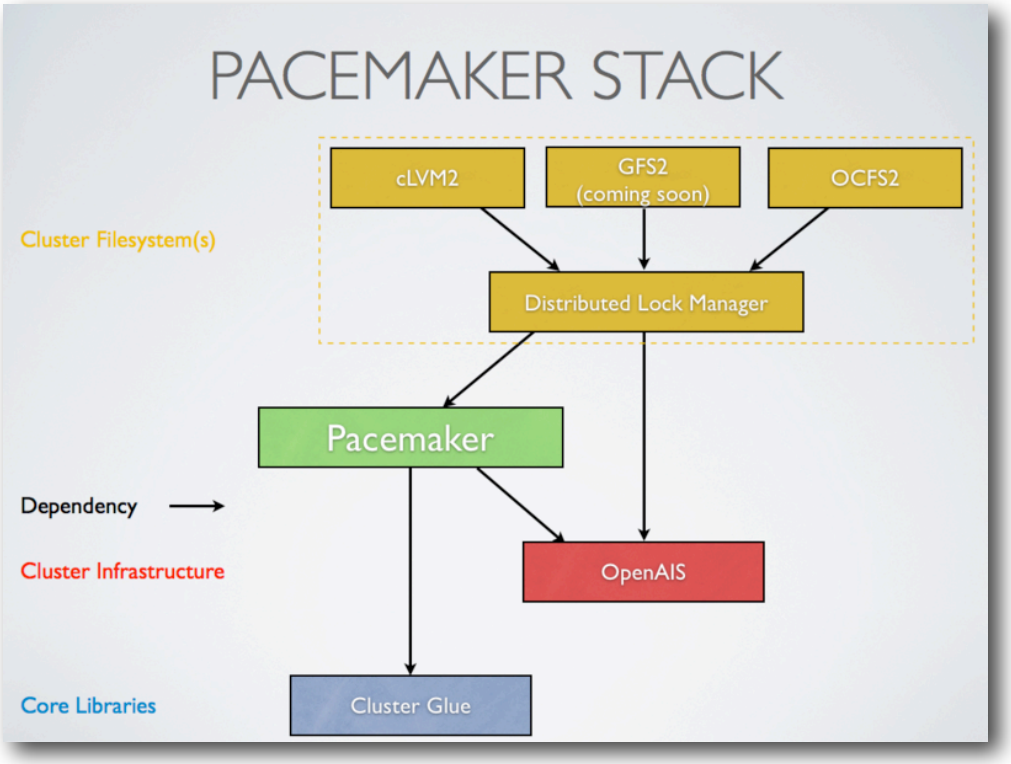
At the highest level, the cluster is made up of three pieces:

- Core cluster infrastructure providing messaging and membership functionality (illustrated in red)
- Non-cluster aware components (illustrated in blue). In a Pacemaker cluster, these pieces include not only the scripts that knows how to start, stop and monitor resources, but also a local daemon that masks the differences between the different standards these scripts implement.
- A brain (illustrated in green) that processes and reacts to events from the cluster (nodes leaving or joining) and resources (eg. monitor failures) as well as configuration changes from the administrator. In response to all of these events, Pacemaker will compute the ideal state of the cluster and plot a path to achieve it. This may include moving resources, stopping nodes and even forcing them offline with remote power switches.



Conceptual overview of the cluster stack

When combined with OpenAIS, Pacemaker also supports popular open source cluster filesystems[1]. Due to recent standardization in the cluster filesystem community, they make use of a common distributed lock manager which makes use of OpenAIS for its messaging capabilities and Pacemaker for its membership (which nodes are up/down) and fencing services.



The Pacemaker stack when running on OpenAIS

[1] Even though Pacemaker also supports Heartbeat, the filesystems need to use the stack for messaging and membership and OpenAIS seems to be what they're standardizing on. Technically it would be possible for them to support Heartbeat as well, however there seems little interest in this.

Installation

OS Installation

Detailed instructions for installing Fedora are available at http://docs.fedoraproject.org/install-guide/f11/ in a number of languages. The abbreviated version is:

Point your browser to http://fedoraproject.org/en/get-fedora-all, locate the **Install Media** section and download the install DVD that matches your hardware.

Burn the disk image to a DVD[2] and boot from it. Or use the image to boot a virtual machine as I have done here. After clicking through the welcome screen, select your language and keyboard layout[3]

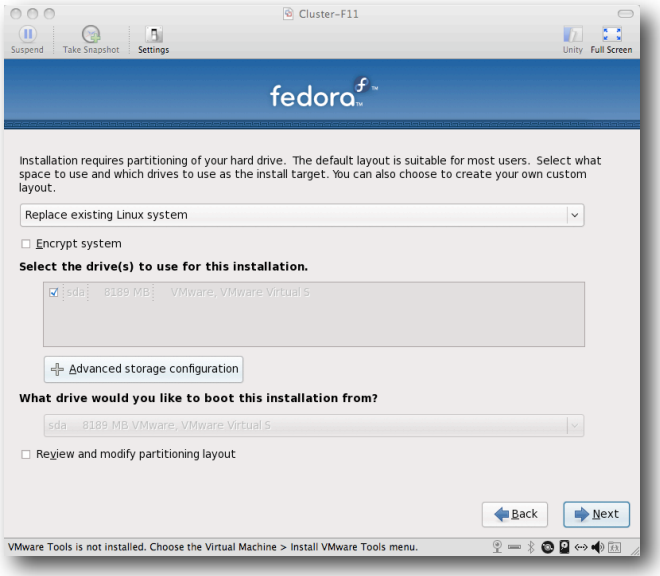
[2] http://docs.fedoraproject.org/readme-burning-isos/

[3] http://docs.fedoraproject.org/install-guide/f11/en-US/html/s1-langselection-x86.html

Assign your machine a host name.[4] I happen to control the clusterlabs.org domain name, so I will use that here.

You will then be prompted to indicate the machine's physical location and to supply a root password[5].

Now select where you want Fedora installed.[6] As I don't care about any existing data, I will accept the default and allow Fedora to use the complete drive.



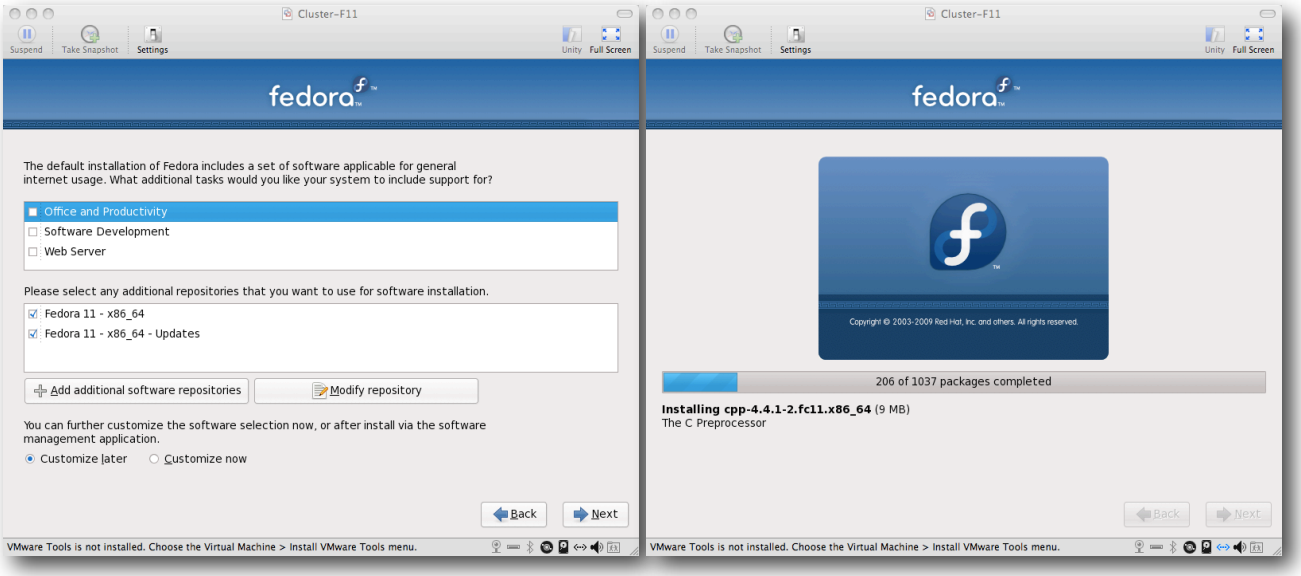
[4] http://docs.fedoraproject.org/install-guide/f11/en-US/html/sn-networkconfig-fedora.html

[5] http://docs.fedoraproject.org/install-guide/f11/en-US/html/sn-account_configuration.html

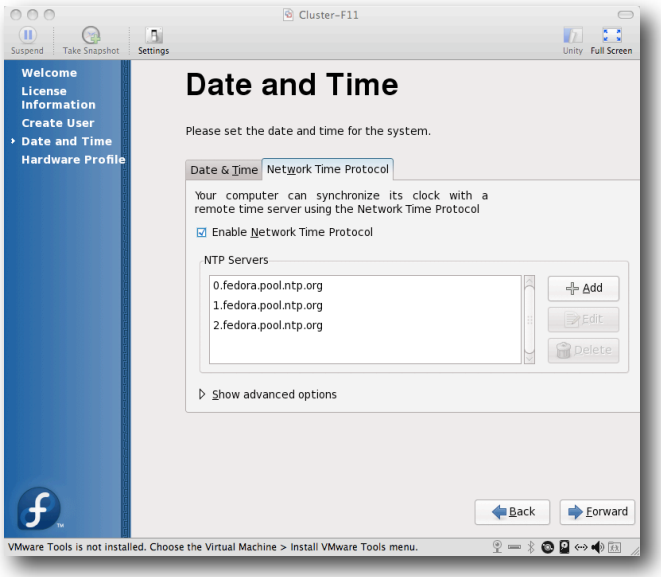
[6] http://docs.fedoraproject.org/install-guide/f11/en-US/html/s1-diskpartsetup-x86.html

The next step is to configure networking. Do not accept the default. Cluster machines should **never** obtain an ip address via DHCP. Here I will use the **internal** addresses for the clusterlab.org network.

Next choose which software should be installed. Deselect the default “Office and Productivity” as its not appropriate for a cluster node. We’ll install any needed software later. After you click next, Fedora will begin installing.



Once the node reboots, follow the on screen instructions [7] to create a system user and configure the time. It is highly recommended to enable NTP on your cluster nodes. Doing so ensures all nodes agree on the current time and makes reading log files significantly easier.



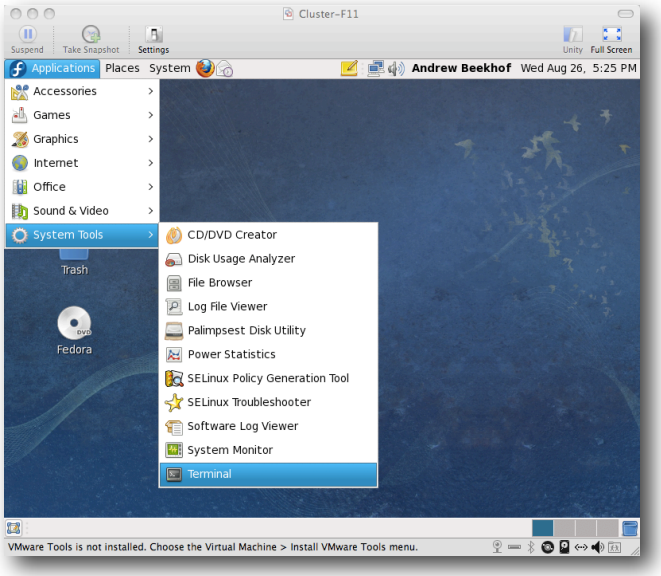
Click through the next screens until you reach the login window. Click on the user you created and supply the password you indicated earlier.

[7] http://docs.fedoraproject.org/install-guide/f11/en-US/html/ch-firstboot.html

Cluster Software Installation

NOTE: Installing the cluster in future versions of Fedora will be significantly easier now that the entire stack has been accepted into the distribution. However for now there are still some hoops to jump through.

Start a terminal by going to Applications -> System Tools -> Terminal



That was the last screenshot by the way, from here on in we're going to be working from the terminal.

Switch to the super user account. You will need to supply the password you entered earlier during the installation process.

```
su -
```

```
[beekhof@test1 ~]$ su -
Password:
[root@test1 ~]#
```

Note that the username (the text before the @ symbol) now indicates we're running as the super user "root".

Tell Fedora Where to Find Pacemaker

Fedora 11 does not ship with Pacemaker, so we need to install it directly from upstream. To do this, we must download and install the YUM repository definition.

```
wget -O /etc/yum.repos.d/pacemaker.repo \
        http://download.opensuse.org/repositories/server:/ha-clustering/Fedora_11/server:ha-clustering.repo
```

```
[root@test1 ~]# wget -O /etc/yum.repos.d/pacemaker.repo http://download.opensuse.org/repositories/server:/
   ha-clustering/Fedora_11/server:ha-clustering.repo
--2009-08-26 17:36:26--  http://download.opensuse.org/repositories/server:/ha-clustering/Fedora_11/
   server:ha-clustering.repo
Resolving download.opensuse.org... 195.135.221.130
Connecting to download.opensuse.org|195.135.221.130|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 310 [application/x-yum-repo]
Saving to: `/etc/yum.repos.d/pacemaker.repo'

100%[======================================>] 310         --.-K/s   in 0s
```

```
2009-08-26 17:36:26 (21.3 MB/s) - `/etc/yum.repos.d/pacemaker.repo' saved [310/310]
```

Before continuing, verify that the repository is installed and that it is providing Pacemaker packages (look for the text highlighted in green).

yum repolist

yum info pacemaker

```
[root@test1 ~]# yum repolist
Loaded plugins: refresh-packagekit
server_ha-clustering                                                    |  951 B     00:00
server_ha-clustering/primary                                            |  11 kB     00:00
server_ha-clustering                                                                 31/31
repo id                 repo name                                                status
fedora                  Fedora 11 - x86_64                                       enabled: 16,577
server_ha-clustering    High Availability/Clustering server technologies (Fedora_11)      enabled:     31
updates                 Fedora 11 - x86_64 - Updates                             enabled:  5,947
repolist: 22,555
[root@test1 ~]# yum info pacemaker
Loaded plugins: refresh-packagekit
Available Packages
Name       : pacemaker
Arch       : i386
Version    : 1.0.5
Release    : 3.1
Size       : 669 k
Repo       : server_ha-clustering
Summary    : Scalable High-Availability cluster resource manager
URL        : http://www.clusterlabs.org
License    : GPLv2+ and LGPLv2+
Description: Pacemaker is an advanced, scalable High-Availability cluster resource
           : manager for Linux-HA (Heartbeat) and/or OpenAIS.
           :
           : It supports "n-node" clusters with significant capabilities for
           : managing resources and dependencies.
           :
           : It will run scripts at initialization, when machines go up or down,
           : when related resources fail and can be configured to periodically check
           : resource health.

Name       : pacemaker
Arch       : x86_64
Version    : 1.0.5
Release    : 3.1
Size       : 698 k
Repo       : server_ha-clustering
Summary    : Scalable High-Availability cluster resource manager
URL        : http://www.clusterlabs.org
License    : GPLv2+ and LGPLv2+
Description: Pacemaker is an advanced, scalable High-Availability cluster resource
           : manager for Linux-HA (Heartbeat) and/or OpenAIS.
           :
           : It supports "n-node" clusters with significant capabilities for
           : managing resources and dependencies.
           :
           : It will run scripts at initialization, when machines go up or down,
           : when related resources fail and can be configured to periodically check
           : resource health.
```

Install the Cluster Software

The tricky part here is to make yum install openais from the *server_ha-clustering* repository by specifying the exact version to install. This is necessary as the version that comes with Fedora is incompatible with the one upstream currently uses to build Pacemaker packages.

yum install -y "openais = 0.80.5" pacemaker

```
[root@test1 ~]# yum install -y "openais = 0.80.5" pacemaker
Loaded plugins: refresh-packagekit
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package openais.x86_64 0:0.80.5-15.1 set to be updated
---> Package pacemaker.x86_64 0:1.0.5-4.1 set to be updated
--> Processing Dependency: pacemaker-libs = 1.0.5-4.1 for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: heartbeat for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: cluster-glue for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: cluster-glue-libs for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libpengine.so.3()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libstonith.so.1()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libpe_rules.so.2()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libhbclient.so.1()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libSaMsg.so.2()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libplumb.so.2()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libcrmcluster.so.1()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libstonithd.so.0()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libcib.so.1()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libpe_status.so.2()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libccmclient.so.1()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libesmtp.so.5()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: liblrm.so.2()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libtransitioner.so.1()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libcrmcommon.so.2()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Processing Dependency: libpils.so.2()(64bit) for package: pacemaker-1.0.5-4.1.x86_64
--> Running transaction check
---> Package cluster-glue.x86_64 0:1.0-12.1 set to be updated
--> Processing Dependency: libopenhpi.so.2()(64bit) for package: cluster-glue-1.0-12.1.x86_64
--> Processing Dependency: libOpenIPMIutils.so.0()(64bit) for package: cluster-glue-1.0-12.1.x86_64
--> Processing Dependency: libOpenIPMIposix.so.0()(64bit) for package: cluster-glue-1.0-12.1.x86_64
--> Processing Dependency: libOpenIPMI.so.0()(64bit) for package: cluster-glue-1.0-12.1.x86_64
---> Package cluster-glue-libs.x86_64 0:1.0-12.1 set to be updated
---> Package heartbeat.x86_64 0:3.0.0-33.2 set to be updated
--> Processing Dependency: PyXML for package: heartbeat-3.0.0-33.2.x86_64
--> Processing Dependency: resource-agents for package: heartbeat-3.0.0-33.2.x86_64
---> Package libesmtp.x86_64 0:1.0.4-10.fc11 set to be updated
---> Package libopenais2.x86_64 0:0.80.5-15.1 set to be updated
---> Package pacemaker-libs.x86_64 0:1.0.5-4.1 set to be updated
--> Running transaction check
---> Package OpenIPMI-libs.x86_64 0:2.0.16-1.fc11 set to be updated
---> Package PyXML.x86_64 0:0.8.4-14 set to be updated
---> Package openhpi-libs.x86_64 0:2.14.0-2.fc11 set to be updated
---> Package resource-agents.x86_64 0:1.0-31.4 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
```

```
 Package                    Arch           Version                Repository                      Size
================================================================================================================
Installing:
 openais                    x86_64         0.80.5-15.1            server_ha-clustering           464 k
 pacemaker                  x86_64         1.0.5-4.1              server_ha-clustering           698 k
Installing for dependencies:
 OpenIPMI-libs              x86_64         2.0.16-1.fc11          fedora                         548 k
 PyXML                      x86_64         0.8.4-14               fedora                         1.1 M
 cluster-glue               x86_64         1.0-12.1               server_ha-clustering           1.4 M
 cluster-glue-libs          x86_64         1.0-12.1               server_ha-clustering           1.2 M
 heartbeat                  x86_64         3.0.0-33.2             server_ha-clustering           1.7 M
 libesmtp                   x86_64         1.0.4-10.fc11          fedora                          62 k
 libopenais2                x86_64         0.80.5-15.1            server_ha-clustering           457 k
 openhpi-libs               x86_64         2.14.0-2.fc11          fedora                         169 k
 pacemaker-libs             x86_64         1.0.5-4.1              server_ha-clustering           2.8 M
 resource-agents            x86_64         1.0-31.4               server_ha-clustering           180 k

Transaction Summary
================================================================================================================
Install     12 Package(s)
Update       0 Package(s)
Remove       0 Package(s)

Total download size: 11 M
Downloading Packages:
(1/12): OpenIPMI-libs-2.0.16-1.fc11.x86_64.rpm                                | 548 kB     00:00
(2/12): PyXML-0.8.4-14.x86_64.rpm                                             | 1.1 MB     00:01
(3/12): cluster-glue-1.0-12.1.x86_64.rpm                                      | 1.4 MB     00:01
(4/12): cluster-glue-libs-1.0-12.1.x86_64.rpm                                 | 1.2 MB     00:01
(5/12): heartbeat-3.0.0-33.2.x86_64.rpm                                       | 1.7 MB     00:01
(6/12): libesmtp-1.0.4-10.fc11.x86_64.rpm                                     |  62 kB     00:00
(7/12): libopenais2-0.80.5-15.1.x86_64.rpm                                    | 457 kB     00:00
(8/12): openais-0.80.5-15.1.x86_64.rpm                                        | 464 kB     00:00
(9/12): openhpi-libs-2.14.0-2.fc11.x86_64.rpm                                 | 169 kB     00:00
(10/12): pacemaker-1.0.5-4.1.x86_64.rpm                                       | 698 kB     00:00
(11/12): pacemaker-libs-1.0.5-4.1.x86_64.rpm                                  | 2.8 MB     00:01
(12/12): resource-agents-1.0-31.4.x86_64.rpm                                  | 180 kB     00:00
----------------------------------------------------------------------------------------------------------------
Total                                                        277 kB/s |  11 MB     00:39
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing     : libopenais2-0.80.5-15.1.x86_64                                         1/12
  Installing     : openhpi-libs-2.14.0-2.fc11.x86_64                                      2/12
  Installing     : libesmtp-1.0.4-10.fc11.x86_64                                          3/12
  Installing     : openais-0.80.5-15.1.x86_64                                             4/12
  Installing     : PyXML-0.8.4-14.x86_64                                                  5/12
  Installing     : OpenIPMI-libs-2.0.16-1.fc11.x86_64                                     6/12
  Installing     : cluster-glue-libs-1.0-12.1.x86_64                                      7/12
  Installing     : cluster-glue-1.0-12.1.x86_64                                           8/12
  Installing     : resource-agents-1.0-31.4.x86_64                                        9/12
  Installing     : heartbeat-3.0.0-33.2.x86_64                                           10/12
  Installing     : pacemaker-libs-1.0.5-4.1.x86_64                                       11/12
  Installing     : pacemaker-1.0.5-4.1.x86_64                                            12/12
```

```
Installed:
  openais.x86_64 0:0.80.5-15.1                      pacemaker.x86_64 0:1.0.5-4.1

Dependency Installed:
  OpenIPMI-libs.x86_64 0:2.0.16-1.fc11              PyXML.x86_64 0:0.8.4-14
  cluster-glue.x86_64 0:1.0-12.1                    cluster-glue-libs.x86_64 0:1.0-12.1
  heartbeat.x86_64 0:3.0.0-33.2                     libesmtp.x86_64 0:1.0.4-10.fc11
  libopenais2.x86_64 0:0.80.5-15.1                  openhpi-libs.x86_64 0:2.14.0-2.fc11
  pacemaker-libs.x86_64 0:1.0.5-4.1                 resource-agents.x86_64 0:1.0-31.4

Complete!
[root@test1 ~]#
```

Prevent OpenAIS Upgrades

We now need to ensure that yum wont install a newer version of openais when we upgrade. To do this, we must install and configure a standard Fedora package called yum-plugin-versionlock.

```
yum install -y yum-plugin-versionlock
rpm -q --qf "%{name}-%{version}-%{release}.%{arch}\n" openais >> /etc/yum/pluginconf.d/versionlock.list
cat /etc/yum/pluginconf.d/versionlock.list
```

```
[root@test1 ~]# yum install -y yum-plugin-versionlock
Loaded plugins: refresh-packagekit
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package yum-plugin-versionlock.noarch 0:1.1.22-1.fc11 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                Arch           Version                Repository     Size
================================================================================
Installing:
 yum-plugin-versionlock noarch         1.1.22-1.fc11          updates        15 k

Transaction Summary
================================================================================
Install      1 Package(s)
Update       0 Package(s)
Remove       0 Package(s)

Total download size: 15 k
Downloading Packages:
yum-plugin-versionlock-1.1.22-1.fc11.noarch.rpm                |  15 kB     00:00
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing     : yum-plugin-versionlock-1.1.22-1.fc11.noarch                1/1

Installed:
  yum-plugin-versionlock.noarch 0:1.1.22-1.fc11
```

```
Complete!
[root@test1 ~]# rpm -q --qf "%{name}-%{version}-%{release}.%{arch}\n" openais >> /etc/yum/pluginconf.d/
   versionlock.list
[root@test1 ~]# cat /etc/yum/pluginconf.d/versionlock.list
openais-0.80.5-15.1.x86_64
[root@test1 ~]#
```

Security Shortcuts

To simplify this guide and focus on the aspects directly connected to clustering, we will now disable the machine's firewall and SELinux installation. Both of these actions create significant security issues and should not be performed on machines that will be exposed to the outside world.

```
sed -i.gres "s/SELINUX=enforcing/SELINUX=permissive/g" /etc/selinux/config
/sbin/chkconfig --del iptables
```

Now reboot all nodes so the new security settings take effect.

Setup

Before You Continue

Repeat the Installation steps so that you have 2 Fedora nodes with the cluster software installed.

For the purposes of this document, the additional node is called *test2* with address 19.168.9.42.

Finalize Node Configuration

Finalize Networking

Confirm that you can communicate with the two new nodes:

ping -c 3 192.168.9.42

```
[root@test1 ~]# ping -c 3 192.168.9.42
PING 192.168.9.42 (192.168.9.42) 56(84) bytes of data.
64 bytes from 192.168.9.42: icmp_seq=1 ttl=64 time=0.343 ms
64 bytes from 192.168.9.42: icmp_seq=2 ttl=64 time=0.402 ms
64 bytes from 192.168.9.42: icmp_seq=3 ttl=64 time=0.558 ms

--- 192.168.9.42 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.343/0.434/0.558/0.092 ms
```

Now we need to make sure we can communicate with the machines by their name. If you have a DNS server, add additional entries for the three machines. Otherwise, you'll need to add the machines to /etc/hosts . Below are the entries for my cluster nodes:

grep test /etc/hosts

```
[root@test1 ~]# grep test /etc/hosts
192.168.9.41 test1.clusterlabs.org test1
192.168.9.42 test2.clusterlabs.org test2
```

We can now verify the setup by again using ping:

ping test2

```
[root@test1 ~]# ping -c 3 test2
PING test2.clusterlabs.org (192.168.9.41) 56(84) bytes of data.
64 bytes from test1.clusterlabs.org (192.168.9.41): icmp_seq=1 ttl=64 time=0.164 ms
64 bytes from test1.clusterlabs.org (192.168.9.41): icmp_seq=2 ttl=64 time=0.475 ms
64 bytes from test1.clusterlabs.org (192.168.9.41): icmp_seq=3 ttl=64 time=0.186 ms

--- test2.clusterlabs.org ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 0.164/0.275/0.475/0.141 ms
```

Configure SSH

SSH is a convenient and secure way to copy files and perform commands remotely. For the purposes of this guide, we will create a key without a password (using the -N "" option) so that we can perform remote actions without being prompted.

NOTE: Unprotected SSH keys, those without a password, are not recommended for servers exposed to the outside world.

Create a new key and allow anyone with that key to log in:

ssh-keygen -t dsa -f ~/.ssh/id_dsa -N ""

cp .ssh/id_dsa.pub .ssh/authorized_keys

```
[root@test1 ~]# ssh-keygen -t dsa -f ~/.ssh/id_dsa -N ""
Generating public/private dsa key pair.
Your identification has been saved in /root/.ssh/id_dsa.
Your public key has been saved in /root/.ssh/id_dsa.pub.
The key fingerprint is:
91:09:5c:82:5a:6a:50:08:4e:b2:0c:62:de:cc:74:44 root@test1.clusterlabs.org
The key's randomart image is:
+--[ DSA 1024]----+
|==.ooEo..        |
|X O + .o o       |
| * A    +        |
|  +      .       |
| .      S        |
|                 |
|                 |
|                 |
|                 |
+-----------------+
[root@test1 ~]# cp .ssh/id_dsa.pub .ssh/authorized_keys
```

Install the key on the other nodes and test that you can now run commands remotely, without being prompted

scp -r .ssh test2:

ssh test2 -- uname -n

```
[root@test1 ~]# scp -r .ssh test2:
The authenticity of host 'test2 (192.168.9.42)' can't be established.
RSA key fingerprint is b1:2b:55:93:f1:d9:52:2b:0f:f2:8a:4e:ae:c6:7c:9a.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'test2,192.168.9.42' (RSA) to the list of known hosts.
root@test2's password:
id_dsa.pub                                                    100%  616     0.6KB/s   00:00
id_dsa                                                        100%  672     0.7KB/s   00:00
known_hosts                                                   100%  400     0.4KB/s   00:00
authorized_keys                                               100%  616     0.6KB/s   00:00
[root@test1 ~]# ssh test2 -- uname -n
test2
[root@test1 ~]#
```

Short Node Names

During installation, we filled in the machine's fully qualifier domain name (FQDN) which can be rather long when it appears in cluster logs and status output. See for yourself how the machine identifies itself:

uname -n

dnsdomainname

```
[root@test1 ~]# uname -n
test1.clusterlabs.org
[root@test1 ~]# dnsdomainname
clusterlabs.org
```

The output from the second command is fine, but we really don't need the domain name included in the basic host details. To address this, we need to update */etc/sysconfig/network*. This is what it should look like before we start.

cat /etc/sysconfig/network

```
NETWORKING=yes
HOSTNAME=test1.clusterlabs.org
GATEWAY=192.168.9.1
```

All we need to do now is strip off the domain name portion, which is stored elsewhere anyway.

sed -i.gres 's/\.[a-z].*//g' /etc/sysconfig/network

```
[root@test1 ~]# sed -i.gres 's/\.[a-z].*//g' /etc/sysconfig/network
```

Now confirm the change was successful. The revised file contents should look something like this.

cat /etc/sysconfig/network

```
[root@test1 ~]# cat /etc/sysconfig/network
NETWORKING=yes
HOSTNAME=test1
GATEWAY=192.168.9.1
```

However we're not finished. The machine wont normally see the shortened host name until about it reboots, but we can force it to update

source /etc/sysconfig/network

hostname $HOSTNAME

```
[root@test1 ~]# source /etc/sysconfig/network
[root@test1 ~]# hostname $HOSTNAME
```

Now check the machine is using the correct names

uname -n

dnsdomainname

```
[root@test1 ~]# uname -n
test1
[root@test1 ~]# dnsdomainname
clusterlabs.org
```

Now repeat on *test2*.

Configuring OpenAIS

Choose a port number and multi-cast[8] address[9].

Be sure that the values you chose do not conflict with any existing clusters you might have. For advice on choosing a multi-cast address, see http://www.29west.com/docs/THPM/multicast-address-assignment.html

For this document, I have chosen port 4000 and used 226.94.1.1 as the multi-cast address.

```
export ais_port=4000
export ais_mcast=226.94.1.1
```

Next we automatically determine the hosts address. By not using the full address, we make the configuration suitable to be copied to other nodes.

```
export ais_addr=`ip addr | grep "inet " | tail -n 1 | awk '{print $4}' | sed s/255/0/`
```

Display and verify the configuration options

```
env | grep ais_
```

```
[root@test1 ~]# env | grep ais_
ais_mcast=226.94.1.1
ais_port=4000
ais_addr=192.168.9.0
```

Once you're happy with the chosen values, update the OpenAIS configuration

```
sed -i.gres "s/.*mcastaddr:.*/mcastaddr:\ $ais_mcast/g" /etc/ais/openais.conf
sed -i.gres "s/.*mcastport:.*/mcastport:\ $ais_port/g" /etc/ais/openais.conf
sed -i.gres "s/.*bindnetaddr:.*/bindnetaddr:\ $ais_addr/g" /etc/ais/openais.conf
```

The final configuration should look something like the sample in the appendix.

Propagate the Configuration

Now we need to copy the changes so far to the other node:

```
for f in /etc/ais/openais.conf /etc/hosts; do scp $f test2:$f ; done
```

```
[root@test1 ~]# for f in /etc/ais/openais.conf /etc/hosts; do scp $f test2:$f ; done
openais.conf                                                  100% 1528     1.5KB/s   00:00
hosts                                                         100%  281     0.3KB/s   00:00
[root@test1 ~]#
```

Verify OpenAIS Installation

Start OpenAIS on the first node

```
/etc/init.d/openais start
```

```
[root@test1 ~]# /etc/init.d/openais start
Starting OpenAIS daemon (aisexec): starting... rc=0: OK
```

[8] http://en.wikipedia.org/wiki/Multicast

[9] http://en.wikipedia.org/wiki/Multicast_address

Check the cluster started correctly and that an initial membership was able to form

grep -e "openais.*network interface" -e "AIS Executive Service" /var/log/messages

grep CLM /var/log/messages

```
[root@test1 ~]# grep -e "openais.*network interface" -e "AIS Executive Service" /var/log/messages
Aug 27 16:23:37 test1 openais[26337]: [MAIN ] AIS Executive Service RELEASE 'subrev 1152 version 0.80'
Aug 27 16:23:38 test1 openais[26337]: [MAIN ] AIS Executive Service: started and ready to provide service.
Aug 27 16:23:38 test1 openais[26337]: [TOTEM] The network interface [192.168.9.41] is now up.
[root@test1 ~]# grep CLM /var/log/messages
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] CLM CONFIGURATION CHANGE
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] New Configuration:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] Members Left:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] Members Joined:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] CLM CONFIGURATION CHANGE
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] New Configuration:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ]     r(0) ip(192.168.9.41)
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] Members Left:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] Members Joined:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ]     r(0) ip(192.168.9.41)
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] got nodejoin message 192.168.9.41
```

With one node functional, its now safe to start OpenAIS on the second node as well.

ssh test2 -- /etc/init.d/openais start

```
[root@test1 ~]# ssh test2 -- /etc/init.d/openais start
Starting OpenAIS daemon (aisexec): starting... rc=0: OK
[root@test1 ~]#
```

Check the cluster formed correctly

grep CPG /var/log/messages

```
[root@test1 ~]# grep CLM /var/log/messages
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] CLM CONFIGURATION CHANGE
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] New Configuration:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] Members Left:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] Members Joined:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] CLM CONFIGURATION CHANGE
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] New Configuration:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ]     r(0) ip(192.168.9.41)
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] Members Left:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] Members Joined:
Aug 27 16:53:15 test1 openais[2166]: [CLM  ]     r(0) ip(192.168.9.41)
Aug 27 16:53:15 test1 openais[2166]: [CLM  ] got nodejoin message 192.168.9.41
Aug 27 16:53:31 test1 openais[2166]: [CLM  ] CLM CONFIGURATION CHANGE
Aug 27 16:53:31 test1 openais[2166]: [CLM  ] New Configuration:
Aug 27 16:53:31 test1 openais[2166]: [CLM  ]     r(0) ip(192.168.9.41)
Aug 27 16:53:31 test1 openais[2166]: [CLM  ] Members Left:
Aug 27 16:53:31 test1 openais[2166]: [CLM  ] Members Joined:
Aug 27 16:53:31 test1 openais[2166]: [CLM  ] CLM CONFIGURATION CHANGE
Aug 27 16:53:31 test1 openais[2166]: [CLM  ] New Configuration:
Aug 27 16:53:31 test1 openais[2166]: [CLM  ]     r(0) ip(192.168.9.41)
Aug 27 16:53:31 test1 openais[2166]: [CLM  ]     r(0) ip(192.168.9.42)
```

```
Aug 27 16:53:31 test1 openais[2166]: [CLM  ] Members Left:
Aug 27 16:53:31 test1 openais[2166]: [CLM  ] Members Joined:
Aug 27 16:53:31 test1 openais[2166]: [CLM  ]     r(0) ip(192.168.9.42)
Aug 27 16:53:31 test1 openais[2166]: [CLM  ] got nodejoin message 192.168.9.41
Aug 27 16:53:31 test1 openais[2166]: [CLM  ] got nodejoin message 192.168.9.42
```

Verify Pacemaker Installation

Now that we have confirmed that OpenAIS is functional we can check the rest of the stack.

grep pcmk_plugin_init /var/log/messages

```
[root@test1 ~]# grep -e "Service initialized" -e pcmk_plugin_init /var/log/messages
Aug 27 16:53:15 test1 openais[2166]: [crm  ] info: pcmk_plugin_init: CRM: Initialized
Aug 27 16:53:15 test1 openais[2166]: [crm  ] Logging: Initialized pcmk_plugin_init
Aug 27 16:53:15 test1 openais[2166]: [crm  ] info: pcmk_plugin_init: Service: 9
Aug 27 16:53:15 test1 openais[2166]: [crm  ] info: pcmk_plugin_init: Local node id: 688498880
Aug 27 16:53:15 test1 openais[2166]: [crm  ] info: pcmk_plugin_init: Local hostname: test1
```

The exact node id will depend on the IP address of the node, unless you also chose 192.168.9.41, you should expect to see a number other than 688498880.

Now verify the Pacemaker processes have been started

ps axf

```
[root@test1 ~]# ps axf
  PID TTY      STAT   TIME COMMAND
    2 ?        S<     0:00 [kthreadd]
    3 ?        S<     0:00  \_ [migration/0]
... lots of processes ...
 2166 pts/0    SLl    0:01 aisexec
 2172 ?        SLs    0:00  \_ /usr/lib64/heartbeat/stonithd
 2173 pts/0    S      0:00  \_ /usr/lib64/heartbeat/cib
 2174 pts/0    S      0:00  \_ /usr/lib64/heartbeat/lrmd
 2175 pts/0    S      0:00  \_ /usr/lib64/heartbeat/attrd
 2176 pts/0    S      0:00  \_ /usr/lib64/heartbeat/pengine
 2177 pts/0    S      0:00  \_ /usr/lib64/heartbeat/crmd
 2178 pts/0    S      0:00  \_ aisexec
```

And finally, check for any ERRORs during startup, there shouldn't be any, and display the cluster's status.

grep ERROR: /var/log/mesages | grep -v unpack_resources

crm_mon

```
[root@test1 ~]# grep ERROR: /var/log/messages | grep -v unpack_resources
[root@test1 ~]# crm_mon
============
Last updated: Thu Aug 27 16:54:55 2009
Stack: openais
Current DC: test1 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
0 Resources configured.
============

Online: [ test1 test2 ]
```

Using Pacemaker Tools

In the dark past, configuring Pacemaker required the administrator to read and write XML. In true UNIX style, there were also a number of different commands that specialized in different aspects of querying and updating the cluster.

Since Pacemaker 1.0, this has all changed and we have an integrated, scriptable, cluster shell that hides all the messy XML scaffolding. It even allows you to queue up several changes at once and commit them atomically.

Take some time to familiarize yourself with what it can do.

crm --help

```
[root@test1 ~]# crm --help

usage:
    crm [-D display_type]
    crm [-D display_type] args
    crm [-D display_type] [-f file]

    Use crm without arguments for an interactive session.
    Supply one or more arguments for a "single-shot" use.
    Specify with -f a file which contains a script. Use '-' for
    standard input or use pipe/redirection.

    crm displays cli format configurations using a color scheme
    and/or in uppercase. Pick one of "color" or "uppercase", or
    use "-D color,uppercase" if you want colorful uppercase.
    Get plain output by "-D plain". The default may be set in
    user preferences (options).

Examples:

    # crm -f stopapp2.cli
    # crm < stopapp2.cli
    # crm resource stop global_www
    # crm status
```

The primary tool for monitoring the status of the cluster is crm_mon (also available as crm status). It can be run in a variety of modes and has a number of output options. To find out about any of the tools that come with Pacemaker, simply invoke them with the --help option or consult the included man pages. Both sets of output are created from the tool, and so will always be in sync with each other and the tool itself.

Additionally, the Pacemaker version and supported cluster stack(s) is available via the --version option.

crm_mon --version

crm_mon --help

```
[root@test1 ~]# crm_mon --version
crm_mon 1.0.5 for OpenAIS and Heartbeat (Build: 462f1569a43740667daf7b0f6b521742e9eb8fa7)

Written by Andrew Beekhof
[root@test1 ~]# crm_mon --help
crm_mon - Provides a summary of cluster's current state.

Outputs varying levels of detail in a number of different formats.
```

```
Usage: crm_mon mode [options]
Options:
 -?, --help            This text
 -$, --version         Version information
 -V, --verbose         Increase debug output

Modes:
 -h, --as-html=value   Write cluster status to the named file
 -w, --web-cgi         Web mode with output suitable for cgi
 -s, --simple-status   Display the cluster status once as a simple one line output (suitable for nagios)
 -S, --snmp-traps=value Send SNMP traps to this station
 -T, --mail-to=value   Send Mail alerts to this user.  See also --mail-from, --mail-host, --mail-prefix

Display Options:
 -n, --group-by-node   Group resources by node
 -r, --inactive  Display inactive resources
 -f, --failcounts      Display resource fail counts
 -o, --operations      Display resource operation history
 -t, --timing-details  Display resource operation history with timing details


Additional Options:
 -i, --interval=value         Update frequency in seconds
 -1, --one-shot        Display the cluster status once on the console and exit
 -N, --disable-ncurses        Disable the use of ncurses
 -d, --daemonize              Run in the background as a daemon
 -p, --pid-file=value         (Advanced) Daemon pid file location
 -F, --mail-from=value        Mail alerts should come from the named user
 -H, --mail-host=value        Mail alerts should be sent via the named host
 -P, --mail-prefix=value      Subjects for mail alerts should start with this string
 -E, --external-agent=value   A program to run when resource operations take place.
 -e, --external-recipient=value      A recipient for your program (assuming you want the program to send
   something to someone).

Examples:

Display the cluster´s status on the console with updates as they occur:

   # crm_mon

Display the cluster´s status on the console just once then exit:

   # crm_mon

Display your cluster´s status, group resources by node, and include inactive resources in the list:

   # crm_mon --group-by-node --inactive

Start crm_mon as a background daemon and have it write the cluster´s status to an HTML file:

   # crm_mon --daemonize --as-html /path/to/docroot/filename.html

Start crm_mon as a background daemon and have it send email alerts:

   # crm_mon --daemonize --mail-to user@example.com --mail-host mail.example.com

Start crm_mon as a background daemon and have it send SNMP alerts:
```

```
    # crm_mon --daemonize --snmp-traps snmptrapd.example.com


Report bugs to pacemaker@oss.clusterlabs.org
```

Create an Active/Passive Cluster

Exploring the Existing Configuration

When Pacemaker starts up, it automatically records the number and details of the nodes in the cluster as well as which stack is being used and the version of Pacemaker being used.

This is what the base configuration should look like.

crm configure show

```
[root@test2 ~]# crm configure show
node test1
node test2
property $id="cib-bootstrap-options" \
    dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
    cluster-infrastructure="openais" \
    expected-quorum-votes="2"
```

For those that are not of afraid of XML, you can see the raw configuration by appending “xml” to the previous command.

crm configure show xml

```
[root@test2 ~]# crm configure show xml
<?xml version="1.0" ?>
<cib admin_epoch="0" crm_feature_set="3.0.1" dc-uuid="test1" epoch="13" have-quorum="1" num_updates="7"
    validate-with="pacemaker-1.0">
  <configuration>
    <crm_config>
      <cluster_property_set id="cib-bootstrap-options">
        <nvpair id="cib-bootstrap-options-dc-version" name="dc-version"
    value="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7"/>
        <nvpair id="cib-bootstrap-options-cluster-infrastructure" name="cluster-infrastructure"
    value="openais"/>
        <nvpair id="cib-bootstrap-options-expected-quorum-votes" name="expected-quorum-votes" value="2"/>
      </cluster_property_set>
    </crm_config>
    <rsc_defaults/>
    <op_defaults/>
    <nodes>
      <node id="test1" type="normal" uname="test1"/>
      <node id="test2" type="normal" uname="test2"/>
    </nodes>
    <resources/>
    <constraints/>
  </configuration>
</cib>
```

The last XML you'll see in this document

Before we make any changes, its a good idea to check the validity of the configuration.

crm_verify -L

```
[root@test1 ~]# crm_verify -L
crm_verify[2195]: 2009/08/27_16:57:12 ERROR: unpack_resources: No STONITH resources have been defined
crm_verify[2195]: 2009/08/27_16:57:12 ERROR: unpack_resources: Either configure some or disable STONITH with
    the stonith-enabled option
crm_verify[2195]: 2009/08/27_16:57:12 ERROR: unpack_resources: NOTE: Clusters with shared data need STONITH
    to ensure data integrity
Errors found during check: config not valid
  -V may provide more details
[root@test1 ~]#
```

As you can see, the tool has found some errors.

In order to guarantee the safety of your data[10], Pacemaker ships with STONITH[11] enabled. However it also knows when no STONITH configuration has been supplied and reports this as a problem (since the cluster would not be able to make progress if a situation requiring node fencing arose).

For now, we will disable this feature and configure it later in the Configuring STONITH section. It is important to note that the use of STONITH is highly encouraged, turning it off tells the cluster to simply pretend that failed nodes are safely powered off. Some vendors will even refuse to support clusters that have it disabled.

To disable STONITH, we set the **stonith-enabled** cluster option to false.

crm configure property stonith-enabled=false

crm_verify -L

```
[root@test1 ~]# crm configure property stonith-enabled=false
[root@test1 ~]# crm_verify -L
[root@test1 ~]#
```

With the new cluster option set, the configuration is now valid.

[10] If the data is corrupt, there is little point in continuing to make it available

[11] A common node fencing mechanism. Used to ensure data integrity by powering off "bad" nodes.

Adding a Resource

The first thing we should do is configure an IP address. Regardless of where the cluster service(s) are running, we need a consistent address to contact them on. Here I will choose and add 192.168.9.101 as the floating address, give it the imaginative name *ClusterIP* and tell the cluster to check that its running every 30 seconds.

Important: The chosen address must not be one already associated with a physical node

```
crm configure primitive ClusterIP ocf:heartbeat:IPaddr2  \
        params ip=192.168.9.101 cidr_netmask=32 \
        op monitor interval=30s
```

The other important piece of information here is *ocf:heartbeat:IPaddr2*. This tells Pacemaker three things about the resource you want to add. The first field, *ocf*, is the standard to which the resource script conforms to and where to find it. The second field is specific to OCF resources and tells the cluster which namespace to find the resource script in, in this case *heartbeat*. The last field indicates the name of the resource script.

To obtain a list of the available resource classes, run

```
crm ra classes
```

```
[root@test1 ~]# crm ra classes
heartbeat
lsb
ocf / heartbeat pacemaker
stonith
```

To then find all the OCF resource agents provided by Pacemaker and Heartbeat, run

```
crm ra list ocf heartbeat
crm ra list ocf pacemaker
```

```
[root@test1 ~]# crm ra list ocf pacemaker
ClusterMon      Dummy           Stateful        SysInfo         SystemHealth    controld
ping            pingd
[root@test1 ~]# crm ra list ocf heartbeat
AoEtarget               AudibleAlarm            ClusterMon              Delay
Dummy                   EvmsSCC                 Evmsd                   Filesystem
ICP                     IPaddr                  IPaddr2                 IPsrcaddr
LVM                     LinuxSCSI               MailTo                  ManageRAID
ManageVE                Pure-FTPd               Raid1                   Route
SAPDatabase             SAPInstance             SendArp                 ServeRAID
SphinxSearchDaemon      Squid                   Stateful                SysInfo
VIPArip                 VirtualDomain           WAS                     WAS6
WinPopup                Xen                     Xinetd                  anything
apache                  db2                     drbd                    eDir88
iSCSILogicalUnit        iSCSITarget             ids                     iscsi
ldirectord              mysql                   mysql-proxy             nfsserver
oracle                  oralsnr                 pgsql                   pingd
portblock               rsyncd                  scsi2reservation        sfex
tomcat                  vmware
[root@test1 ~]#
```

Now verify that the IP resource has been added and display the cluster's status to see that it is now active.

crm configure show

crm_mon

```
[root@test1 ~]# crm configure show
node test1
node test2
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
property $id="cib-bootstrap-options" \
    dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
    cluster-infrastructure="openais" \
    expected-quorum-votes="2" \
    stonith-enabled="false" \
[root@test1 ~]# crm_mon
============
Last updated: Fri Aug 28 15:23:48 2009
Stack: openais
Current DC: test1 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
1 Resources configured.
============

Online: [ test1 test2 ]

ClusterIP (ocf::heartbeat:IPaddr):   Started test1
```

Perform a Failover

Being a high-availability cluster, we should test failover of our new resource before moving on.

First, find the node on which the IP address is running.

crm resource status ClusterIP

```
[root@test1 ~]# crm resource status ClusterIP
resource ClusterIP is running on: test1
[root@test1 ~]#
```

Shut down OpenAIS on that machine.

ssh test1 -- /etc/init.d/openais stop

```
[root@test1 ~]# ssh test1 -- /etc/init.d/openais stop
Stopping OpenAIS daemon (aisexec): ........OK
[root@test1 ~]#
```

Once OpenAIS is no longer running, go to the other node and check the cluster status with crm_mon.

```
[root@test2 ~]# crm_mon
============
Last updated: Fri Aug 28 15:27:35 2009
Stack: openais
Current DC: test2 - partition WITHOUT quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
1 Resources configured.
============

Online: [ test2 ]
OFFLINE: [ test1 ]
```

There are three things to notice about the cluster's current state. The first is that, as expected, test1 is now offline. However we can also see that *ClusterIP* isn't running anywhere!

Quorum and Two-Node Clusters

This is because the cluster no longer has quorum, as can be seen by the text "partition WITHOUT quorum" (highlighted green) in the output above. In order to reduce the possibility of data corruption, Pacemaker's default behavior is to stop all resources if the cluster does not have quorum.

A cluster is said to have quorum when more than half the known or expected nodes are online, or for the mathematically inclined, whenever the following equation is true:

$$total\_nodes - 1 < 2 * active\_nodes$$

Therefore a two-node cluster only has quorum when both nodes are running, which is no longer the case for our cluster. This would normally make the creation of a two-node cluster pointless[12], however it is possible to control how Pacemaker behaves when quorum is lost. In particular, we can tell the cluster to simply ignore quorum altogether.

crm configure property no-quorum-policy=ignore

crm configure show

```
[root@test1 ~]# crm configure property no-quorum-policy=ignore
[root@test1 ~]# crm configure show
node test1
node test2
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
property $id="cib-bootstrap-options" \
    dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
    cluster-infrastructure="openais" \
    expected-quorum-votes="2" \
    stonith-enabled="false" \
    no-quorum-policy="ignore"
```

[12] Actually some would argue that two-node clusters are always pointless, but that is an argument for another time.

After a few moments, the cluster will start the IP address on the remaining node. Note that the cluster still does not have quorum.

```
[root@test2 ~]# crm_mon
============
Last updated: Fri Aug 28 15:30:18 2009
Stack: openais
Current DC: test2 - partition WITHOUT quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
1 Resources configured.
============

Online: [ test2 ]
OFFLINE: [ test1 ]

ClusterIP (ocf::heartbeat:IPaddr):   Started test2
```

Now simulate node recovery by restarting the cluster stack on test1 and check the cluster's status.

/etc/init.d/openais start

crm_mon

```
[root@test1 ~]# /etc/init.d/openais start
Starting OpenAIS daemon (aisexec): starting... rc=0: OK
[root@test1 ~]# crm_mon
============
Last updated: Fri Aug 28 15:32:13 2009
Stack: openais
Current DC: test2 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
1 Resources configured.
============

Online: [ test1 test2 ]

ClusterIP (ocf::heartbeat:IPaddr):        Started test1
```

Here we see something that some may consider surprising, the IP is back running at its original location!

Prevent Resources from Moving after Recovery

In some circumstances it is highly desirable to prevent healthy resources from being moved around the cluster. Move resources almost always requires a period of downtime and for complex services like Oracle databases, this period can be quite long.

To address this, Pacemaker has the concept of resource stickiness which controls how much a service prefers to stay running where it is. You may like to think of it as the "cost" of any downtime. By default, Pacemaker assumes there is zero cost associated with moving resources and will do so to achieve "optimal[13]" resource placement. We can specify a different stickiness for every resource, but it is often sufficient to change the default.

crm configure rsc_defaults resource-stickiness=100

crm configure show

```
[root@test2 ~]# crm configure show
node test1
node test2
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
property $id="cib-bootstrap-options" \
    dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
    cluster-infrastructure="openais" \
    expected-quorum-votes="2" \
    stonith-enabled="false" \
    no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
    resource-stickiness="100"
```

If we now retry the failover test, we see that as expected *ClusterIP* still moves to *test2* when *test1* is taken offline.

ssh test1 -- /etc/init.d/openais stop

ssh test2 -- crm_mon -1

```
[root@test1 ~]# ssh test1 -- /etc/init.d/openais stop
Stopping OpenAIS daemon (aisexec): ........OK
[root@test1 ~]# ssh test2 -- crm_mon -1
============
Last updated: Fri Aug 28 15:39:38 2009
Stack: openais
Current DC: test2 - partition WITHOUT quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
1 Resources configured.
============

Online: [ test2 ]
OFFLINE: [ test1 ]

ClusterIP (ocf::heartbeat:IPaddr):   Started test2
```

[13] It should be noted that Pacemaker's definition of optimal may not always agree with that of a human's. The order in which Pacemaker processes lists of resources and nodes create implicit preferences (required in order to create a stabile solution) in situations where the administrator had not explicitly specified some.

However when we bring *test1* back online, *ClusterIP* now remains running on *test2*.

```
[root@test1 ~]# /etc/init.d/openais start
Starting OpenAIS daemon (aisexec): starting... rc=0: OK
[root@test1 ~]# crm_mon
============
Last updated: Fri Aug 28 15:41:23 2009
Stack: openais
Current DC: test2 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
1 Resources configured.
============

Online: [ test1 test2 ]

ClusterIP (ocf::heartbeat:IPaddr):        Started test2
```

Apache - Adding More Services

Now that we have a basic but functional active/passive two-node cluster, we're ready to add some real services. We're going to start with Apache because its a feature of many clusters and relatively simple to configure.

Preparation

First we need to create a page for Apache to serve up. On Fedora 11 the default Apache docroot is */var/www/html*, so we'll create an index file there.

```
cat <<-END >/var/www/html/index.html
<html>
<body>My Test Site - test1</body>
</html>
END
```

```
[root@test1 ~]# cat <<-END >/var/www/html/index.html
> <html>
> <body>My Test Site - test1</body>
> </html>
> END
[root@test1 ~]#
```

For the moment, we will simplify things by serving up only a static site and manually sync the data between the two nodes. So run the command again on test2.

```
cat <<-END >/var/www/html/index.html
<html>
<body>My Test Site - test2</body>
</html>
END
```

```
[root@test2 ~]# cat <<-END >/var/www/html/index.html
> <html>
> <body>My Test Site - test2</body>
> </html>
> END
[root@test2 ~]#
```

Update the Configuration

At this point, Apache is ready to go, all that needs to be done is to add it to the cluster. Lets call the resource WebSite. We need to use an OCF script called *apache* in the *heartbeat* namespace[14], the only required parameter is the path to the main Apache configuration file and we'll tell the cluster to check once a minute that apache is still running.

```
crm configure primitive WebSite ocf:heartbeat:apache \
        params configfile=/etc/httpd/conf/httpd.conf \
        op monitor interval=1min
crm configure show
crm_mon
```

```
[root@test1 ~]# crm configure primitive WebSite ocf:heartbeat:apache params configfile=/etc/httpd/conf/
   httpd.conf op monitor interval=1min
[root@test1 ~]# crm configure show
node test1
node test2
primitive WebSite ocf:heartbeat:apache \
   params configfile="/etc/httpd/conf/httpd.conf" \
   op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
   params ip="192.168.9.101" cidr_netmask="32" \
   op monitor interval="30s"
property $id="cib-bootstrap-options" \
   dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
   cluster-infrastructure="openais" \
   expected-quorum-votes="2" \
   stonith-enabled="false" \
   no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
   resource-stickiness="100"
```

After a short delay, we should see the cluster start apache

```
[root@test1 ~]# crm_mon
============
Last updated: Fri Aug 28 16:12:49 2009
Stack: openais
Current DC: test2 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
2 Resources configured.
============

Online: [ test1 test2 ]

ClusterIP (ocf::heartbeat:IPaddr):   Started test2
WebSite   (ocf::heartbeat:apache):   Started test1
```

Wait a moment, the WebSite resource isn't running on the same host as our IP address!

[14] Compare the key used here *ocf:heartbeat:apache* with the one we used earlier for the IP address: *ocf:heartbeat:IPaddr2*

Ensuring Resources Run on the Same Host

To reduce the load on any one machine, Pacemaker will generally try to spread the configured resources across the cluster nodes. However we can tell the cluster that two resources are related and need to run on the same host (or not at all). Here we instruct the cluster that *WebSite* can only run on the host that *ClusterIP* is active on. If *ClusterIP* is not active anywhere, *WebSite* will not be permitted to run anywhere.

crm configure colocation website-with-ip INFINITY: WebSite ClusterIP

crm configure show

crm_mon

```
[root@test1 ~]# crm configure colocation website-with-ip INFINITY: WebSite ClusterIP
[root@test1 ~]# crm configure show
node test1
node test2
primitive WebSite ocf:heartbeat:apache \
    params configfile="/etc/httpd/conf/httpd.conf" \
    op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
colocation website-with-ip inf: WebSite ClusterIP
property $id="cib-bootstrap-options" \
    dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
    cluster-infrastructure="openais" \
    expected-quorum-votes="2" \
    stonith-enabled="false" \
    no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
    resource-stickiness="100"
[root@test1 ~]# crm_mon
============
Last updated: Fri Aug 28 16:14:34 2009
Stack: openais
Current DC: test2 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
2 Resources configured.
============

Online: [ test1 test2 ]

ClusterIP (ocf::heartbeat:IPaddr):   Started test2
WebSite   (ocf::heartbeat:apache):   Started test2
```

Controlling Resource Start/Stop Ordering

When Apache starts, it binds to the available IP addresses. It doesn't know about any addresses we add afterwards, so not only do they need to run on the same node, but we need to make sure *ClusterIP* is already active before we start *WebSite*. We do this by adding an ordering constraint. We need to give it a name (chose something descriptive like *apache-after-ip*), indicate that its mandatory (so that any recovery for *ClusterIP* will also trigger recovery of *WebSite*) and list the two resources in the order we need them to start.

crm configure order apache-after-ip mandatory: ClusterIP WebSite

crm configure show

```
[root@test1 ~]# crm configure order apache-after-ip mandatory: ClusterIP WebSite
[root@test1 ~]# crm configure show
node test1
node test2
primitive WebSite ocf:heartbeat:apache \
    params configfile="/etc/httpd/conf/httpd.conf" \
    op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
colocation website-with-ip inf: WebSite ClusterIP
order apache-after-ip inf: ClusterIP WebSite
property $id="cib-bootstrap-options" \
    dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
    cluster-infrastructure="openais" \
    expected-quorum-votes="2" \
    stonith-enabled="false" \
    no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
    resource-stickiness="100"
```

Specifying a Preferred Location

Pacemaker does not rely on any sort of hardware symmetry between nodes, so it may well be that one machine is more powerful than the other. In such cases it makes sense to host the resources there if it is available. To do this we create a location constraint. Again we give it a descriptive name (*prefer-test1*), specify the resource we want to run there (*WebSite*), how badly we'd like it to run there (we'll use *50* for now, but in a two-node situation almost any value above 0 will do) and the host's name.

crm configure location prefer-test1 WebSite rule 50: test1

crm configure show

crm_mon

```
[root@test1 ~]# crm configure location prefer-test1 WebSite 50: test1
[root@test1 ~]# crm configure show
node test1
node test2
primitive WebSite ocf:heartbeat:apache \
   params configfile="/etc/httpd/conf/httpd.conf" \
   op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
   params ip="192.168.9.101" cidr_netmask="32" \
   op monitor interval="30s"
location prefer-test1 WebSite 50: test1
colocation website-with-ip inf: WebSite ClusterIP
property $id="cib-bootstrap-options" \
   dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
   cluster-infrastructure="openais" \
   expected-quorum-votes="2" \
   stonith-enabled="false" \
   no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
   resource-stickiness="100"
[root@test1 ~]# crm_mon
============
Last updated: Fri Aug 28 16:17:35 2009
Stack: openais
Current DC: test2 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
2 Resources configured.
============

Online: [ test1 test2 ]

ClusterIP (ocf::heartbeat:IPaddr):   Started test2
WebSite   (ocf::heartbeat:apache):   Started test2
```

Wait a minute, the resources are still on test2!

Even though we now prefer test1 over test2, that preference is (intentionally) less than the resource stickiness (how much we preferred not to have unnecessary downtime).

To see the current placement scores, you can use a tool called ptest

ptest -sL

There is a way to force them to move though...

Manually Moving Resources Around the Cluster

There are always times when an administrator needs to override the cluster and force resources to move to a specific location. Underneath we use location constraints like the one we created above, happily you don't need to care. Just provide the name of the resource and the intended location, we'll do the rest.

crm resource move WebSite test1

crm_mon

```
[root@test1 ~]# crm resource move WebSite test1
[root@test1 ~]# crm_mon
============
Last updated: Fri Aug 28 16:19:24 2009
Stack: openais
Current DC: test2 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
2 Resources configured.
============

Online: [ test1 test2 ]

ClusterIP (ocf::heartbeat:IPaddr):   Started test1
WebSite   (ocf::heartbeat:apache):   Started test1
```

Notice how the colocation rule we created has ensured that *ClusterIP* was also moved to *test1*.

For the curious, we can see the effect of this command by examining the configuration

crm configure show

```
[root@test1 ~]# crm configure show
node test1
node test2
primitive WebSite ocf:heartbeat:apache \
    params configfile="/etc/httpd/conf/httpd.conf" \
    op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
location cli-prefer-WebSite WebSite \
    rule $id="cli-prefer-rule-WebSite" inf: #uname eq test1
location prefer-test1 WebSite 50: test1
colocation website-with-ip inf: WebSite ClusterIP
property $id="cib-bootstrap-options" \
    dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
    cluster-infrastructure="openais" \
    expected-quorum-votes="2" \
    stonith-enabled="false" \
    no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
    resource-stickiness="100"
```

Highlighted is the automated constraint used to move the resources to test1

Giving Control Back to the Cluster

Once we've finished whatever activity that required us to move the resources to test1, in our case nothing, we can then allow the cluster to resume normal operation with the **unmove** command. Since we previously configured a default stickiness, the resources will remain on *test1*.

crm resource unmove WebSite

crm configure show

```
[root@test1 ~]# crm resource unmove WebSite
[root@test1 ~]# crm configure show
node test1
node test2
primitive WebSite ocf:heartbeat:apache \
   params configfile="/etc/httpd/conf/httpd.conf" \
   op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
   params ip="192.168.9.101" cidr_netmask="32" \
   op monitor interval="30s"
location prefer-test1 WebSite 50: test1
colocation website-with-ip inf: WebSite ClusterIP
property $id="cib-bootstrap-options" \
   dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
   cluster-infrastructure="openais" \
   expected-quorum-votes="2" \
   stonith-enabled="false" \
   no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
   resource-stickiness="100"
```

Note that the automated constraint is now gone. If we check the cluster status, we can also see that as expected the resources are still active on *test1*.

```
[root@test1 ~]# crm_mon
============
Last updated: Fri Aug 28 16:20:53 2009
Stack: openais
Current DC: test2 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
2 Resources configured.
============

Online: [ test1 test2 ]

ClusterIP (ocf::heartbeat:IPaddr):   Started test1
WebSite   (ocf::heartbeat:apache):   Started test1
```

Using DRBD for Shared Storage

Even if you're serving up static websites, having to manually synchronize the contents of that website to all the machines in the cluster is not ideal. For dynamic websites, such as a wiki, its not even an option. Not everyone care afford network-attached storage but somehow the data needs to be kept in sync. Enter DRBD which can be thought of as network based RAID-1. See http://www.drbd.org/ for more details.

Install Pre-requisites

DRBD does not currently ship with Fedora and since there is a kernel component, can be sensitive to system updates which may change the kernel's APIs and ABIs. For this reason we'll simply build our own DRBD packages - to be sure they are a perfect match for the machine.

First we need to install a few packages that DRBD needs:

yum install -y flex gcc glibc-devel kernel-headers kernel-devel rpm-build

```
[root@test1 ~]# yum install -y flex gcc glibc-devel kernel-headers kernel-devel rpm-build
Loaded plugins: refresh-packagekit, versionlock
Reading version lock configuration
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package flex.x86_64 0:2.5.35-5.fc11 set to be updated
---> Package gcc.x86_64 0:4.4.1-2.fc11 set to be updated
--> Processing Dependency: cloog-ppl >= 0.15 for package: gcc-4.4.1-2.fc11.x86_64
---> Package glibc-devel.x86_64 0:2.10.1-4 set to be updated
--> Processing Dependency: glibc-headers = 2.10.1-4 for package: glibc-devel-2.10.1-4.x86_64
---> Package kernel-devel.x86_64 0:2.6.29.6-217.2.16.fc11 set to be installed
---> Package kernel-headers.x86_64 0:2.6.29.6-217.2.16.fc11 set to be updated
---> Package rpm-build.x86_64 0:4.7.1-1.fc11 set to be updated
--> Processing Dependency: elfutils >= 0.128 for package: rpm-build-4.7.1-1.fc11.x86_64
--> Processing Dependency: xz for package: rpm-build-4.7.1-1.fc11.x86_64
--> Running transaction check
---> Package cloog-ppl.x86_64 0:0.15-0.9.gitb9d79.fc11 set to be updated
--> Processing Dependency: libppl.so.7()(64bit) for package: cloog-ppl-0.15-0.9.gitb9d79.fc11.x86_64
--> Processing Dependency: libppl_c.so.2()(64bit) for package: cloog-ppl-0.15-0.9.gitb9d79.fc11.x86_64
---> Package elfutils.x86_64 0:0.142-1.fc11 set to be updated
--> Processing Dependency: elfutils-libs-x86_64 = 0.142-1.fc11 for package: elfutils-0.142-1.fc11.x86_64
--> Processing Dependency: libdw.so.1(ELFUTILS_0.127)(64bit) for package: elfutils-0.142-1.fc11.x86_64
--> Processing Dependency: libasm.so.1(ELFUTILS_1.0)(64bit) for package: elfutils-0.142-1.fc11.x86_64
--> Processing Dependency: libdw.so.1(ELFUTILS_0.126)(64bit) for package: elfutils-0.142-1.fc11.x86_64
--> Processing Dependency: libdw.so.1(ELFUTILS_0.138)(64bit) for package: elfutils-0.142-1.fc11.x86_64
--> Processing Dependency: libdw.so.1(ELFUTILS_0.122)(64bit) for package: elfutils-0.142-1.fc11.x86_64
--> Processing Dependency: libdw.so.1()(64bit) for package: elfutils-0.142-1.fc11.x86_64
--> Processing Dependency: libasm.so.1()(64bit) for package: elfutils-0.142-1.fc11.x86_64
```

```
---> Package glibc-headers.x86_64 0:2.10.1-4 set to be updated
---> Package xz.x86_64 0:4.999.8-0.8.beta.20090817git.fc11 set to be updated
--> Running transaction check
---> Package elfutils-libs.x86_64 0:0.142-1.fc11 set to be updated
---> Package ppl.x86_64 0:0.10.2-3.fc11 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package             Arch        Version                                 Repository    Size
================================================================================
Installing:
 flex                x86_64      2.5.35-5.fc11                           fedora       319 k
 gcc                 x86_64      4.4.1-2.fc11                            updates       11 M
 glibc-devel         x86_64      2.10.1-4                                updates      998 k
 kernel-devel        x86_64      2.6.29.6-217.2.16.fc11                  updates      6.2 M
 kernel-headers      x86_64      2.6.29.6-217.2.16.fc11                  updates      855 k
 rpm-build           x86_64      4.7.1-1.fc11                            updates      128 k
Installing for dependencies:
 cloog-ppl           x86_64      0.15-0.9.gitb9d79.fc11                  updates       90 k
 elfutils            x86_64      0.142-1.fc11                            updates      232 k
 elfutils-libs       x86_64      0.142-1.fc11                            updates      205 k
 glibc-headers       x86_64      2.10.1-4                                updates      630 k
 ppl                 x86_64      0.10.2-3.fc11                           updates      1.7 M
 xz                  x86_64      4.999.8-0.8.beta.20090817git.fc11       updates      123 k

Transaction Summary
================================================================================
Install     12 Package(s)
Update       0 Package(s)
Remove       0 Package(s)

Total size: 22 M
Downloading Packages:
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing     : elfutils-libs-0.142-1.fc11.x86_64                       1/12
  Installing     : elfutils-0.142-1.fc11.x86_64                            2/12
  Installing     : xz-4.999.8-0.8.beta.20090817git.fc11.x86_64             3/12
  Installing     : ppl-0.10.2-3.fc11.x86_64                                4/12
  Installing     : cloog-ppl-0.15-0.9.gitb9d79.fc11.x86_64                 5/12
  Installing     : kernel-headers-2.6.29.6-217.2.16.fc11.x86_64            6/12
  Installing     : rpm-build-4.7.1-1.fc11.x86_64                           7/12
  Installing     : flex-2.5.35-5.fc11.x86_64                               8/12
  Installing     : glibc-headers-2.10.1-4.x86_64                           9/12
  Installing     : glibc-devel-2.10.1-4.x86_64                            10/12
  Installing     : kernel-devel-2.6.29.6-217.2.16.fc11.x86_64             11/12
  Installing     : gcc-4.4.1-2.fc11.x86_64                                12/12

Installed:
  flex.x86_64 0:2.5.35-5.fc11                   gcc.x86_64 0:4.4.1-2.fc11
  glibc-devel.x86_64 0:2.10.1-4                 kernel-devel.x86_64 0:2.6.29.6-217.2.16.fc11
  kernel-headers.x86_64 0:2.6.29.6-217.2.16.fc11 rpm-build.x86_64 0:4.7.1-1.fc11
```

```
Dependency Installed:
  cloog-ppl.x86_64 0:0.15-0.9.gitb9d79.fc11   elfutils.x86_64 0:0.142-1.fc11
  elfutils-libs.x86_64 0:0.142-1.fc11         glibc-headers.x86_64 0:2.10.1-4
  ppl.x86_64 0:0.10.2-3.fc11                  xz.x86_64 0:4.999.8-0.8.beta.20090817git.fc11

Complete!
[root@test2 ~]#
```

Build DRBD Packages

Once the development packages are installed, we can begin building DRBD[15].

wget http://oss.linbit.com/drbd/8.3/drbd-8.3.2.tar.gz

tar zxf drbd-8.3.2.tar.gz

cd drbd-8.3.2

sed -i.gres "s/%files\ -f\ %{_builddir}\/%{name}-%{version}\//%files\ -f\ /g" drbd.spec.in

grep files.*file.list *spec.in

make rpm

```
[root@test1 ~]# wget http://oss.linbit.com/drbd/8.3/drbd-8.3.2.tar.gz
--2009-08-31 10:58:39--  http://oss.linbit.com/drbd/8.3/drbd-8.3.2.tar.gz
Resolving oss.linbit.com... 212.69.161.111
Connecting to oss.linbit.com|212.69.161.111|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 415152 (405K) [application/x-gzip]
Saving to: `drbd-8.3.2.tar.gz'

100%[======================================>] 415,152     1.05M/s   in 0.4s

2009-08-31 10:58:39 (1.05 MB/s) - `drbd-8.3.2.tar.gz' saved [415152/415152]

[root@test1 ~]# tar zxf drbd-8.3.2.tar.gz
[root@test1 ~]# cd drbd-8.3.2
[root@test1 ~]# sed -i.gres "s/%files\ -f\ %{_builddir}\/%{name}-%{version}\//%files\ -f\ /g" drbd.spec.in
[root@test1 drbd-8.3.2]# grep files.*file.list *spec.in
%files -f file.list
```

The following build log is quite long and is only included for reference in case it does not work on your machine. Most people can skip to the end.

```
[root@test1 drbd-8.3.2]# make rpm
checking for presence of 8\.3\.2 in various changelog files

   debian/changelog needs update [ignored]

make -C drbd drbd_buildtag.c
make[1]: Entering directory `/root/drbd-8.3.2/drbd'
make[1]: Leaving directory `/root/drbd-8.3.2/drbd'
make -C documentation doc
make[1]: Entering directory `/root/drbd-8.3.2/documentation'
make[1]: Nothing to be done for `doc'.
```

[15] At the time of writing, the latest version was 8.3.2. If a later version is now available it would be advisable to try that first.

```
make[1]: Leaving directory `/root/drbd-8.3.2/documentation'
test -e .filelist
ln -sf drbd/linux/drbd_config.h drbd_config.h
rm -f drbd-8.3.2
ln -s . drbd-8.3.2
for f in $(<.filelist) ; do [ -e $f ] && continue ; echo missing: $f ; exit 1; done
grep debian .filelist >/dev/null 2>&1 && _DEB=-debian || _DEB="" ; \
   tar --owner=0 --group=0 -czf - -T .filelist > drbd-8.3.2$_DEB.tar.gz
rm drbd-8.3.2
mkdir -p dist/BUILD \
            dist/RPMS  \
            dist/SPECS \
            dist/SOURCES \
            dist/TMP \
            dist/install \
            dist/SRPMS
[ -h dist/SOURCES/drbd-8.3.2.tar.gz ] || \
     ln -s /root/drbd-8.3.2/drbd-8.3.2.tar.gz \
             /root/drbd-8.3.2/dist/SOURCES/drbd-8.3.2.tar.gz
if test drbd.spec.in -nt dist/SPECS/drbd.spec ; then \
      sed -e "s/^\(Version:\).*/\1 8.3.2/;" \
          -e "s/^\(Packager:\).*/\1 root@test1/;" < drbd.spec.in \
      > dist/SPECS/drbd.spec ; \
   fi
rpmbuild -bb \
       --define "_topdir /root/drbd-8.3.2/dist" \
       --define "buildroot /root/drbd-8.3.2/dist/install" \
       --define "kernelversion 2.6.29.6-217.2.16.fc11.x86_64" \
       --define "kdir /lib/modules/2.6.29.6-217.2.16.fc11.x86_64/build" \
        \
       /root/drbd-8.3.2/dist/SPECS/drbd.spec
Executing(%prep): /bin/sh -e /var/tmp/rpm-tmp.uU3DHT
+ umask 022
+ cd /root/drbd-8.3.2/dist/BUILD
+ cd /root/drbd-8.3.2/dist/BUILD
+ rm -rf drbd-8.3.2
+ /usr/bin/gzip -dc /root/drbd-8.3.2/dist/SOURCES/drbd-8.3.2.tar.gz
+ /bin/tar -xvvf -
-rw-rw-r-- root/root       875 2008-11-24 11:43 drbd-8.3.2/.gitignore
drwxr-xr-x                  Creating directory: drbd-8.3.2
-rw-rw-r-- root/root     17990 2008-11-24 11:43 drbd-8.3.2/COPYING
-rw-rw-r-- root/root     25537 2009-07-03 13:39 drbd-8.3.2/ChangeLog
-rw-rw-r-- root/root      7054 2009-05-26 14:49 drbd-8.3.2/Makefile
-rw-rw-r-- root/root       425 2009-05-26 14:49 drbd-8.3.2/README
-rw-rw-r-- root/root      2122 2009-05-26 14:49 drbd-8.3.2/ROADMAP
-rw-rw-r-- root/root       191 2009-05-26 14:49 drbd-8.3.2/benchmark/Makefile
drwxr-xr-x                  Creating directory: drbd-8.3.2/benchmark
-rw-rw-r-- root/root        95 2007-10-22 17:38 drbd-8.3.2/benchmark/README
-rw-rw-r-- root/root     10173 2008-11-24 11:43 drbd-8.3.2/benchmark/dm.c
-rw-rw-r-- root/root      8639 2008-11-24 11:43 drbd-8.3.2/benchmark/io-latency-test.c
-rw-rw-r-- root/root      3770 2009-05-26 14:49 drbd-8.3.2/documentation/Makefile
drwxr-xr-x                  Creating directory: drbd-8.3.2/documentation
-rw-rw-r-- root/root      2860 2008-11-24 11:43 drbd-8.3.2/documentation/Makefile.lang
-rw-rw-r-- root/root     57228 2009-06-25 10:28 drbd-8.3.2/documentation/drbd.conf.sgml
-rw-rw-r-- root/root      3031 2009-06-09 13:33 drbd-8.3.2/documentation/drbd.sgml
-rw-rw-r-- root/root     16524 2009-06-25 10:28 drbd-8.3.2/documentation/drbdadm.sgml
-rw-rw-r-- root/root      2750 2009-05-26 14:49 drbd-8.3.2/documentation/drbddisk.sgml
```

```
-rw-rw-r-- root/root      6353 2009-06-09 13:33 drbd-8.3.2/documentation/drbdmeta.sgml
-rw-rw-r-- root/root     60284 2009-06-09 13:33 drbd-8.3.2/documentation/drbdsetup.sgml
-rw-rw-r-- root/root     27389 2009-08-31 13:08 drbd-8.3.2/drbd.spec.in
-rw-rw-r-- root/root      1734 2009-06-16 14:24 drbd-8.3.2/drbd/Kconfig
drwxr-xr-x                  Creating directory: drbd-8.3.2/drbd
-rw-rw-r-- root/root      7094 2009-06-16 15:12 drbd-8.3.2/drbd/Makefile
-rw-rw-r-- root/root       377 2009-06-09 13:33 drbd-8.3.2/drbd/Makefile-2.6
-rw-rw-r-- root/root      4932 2009-05-26 14:49 drbd-8.3.2/drbd/cn_queue.c
-rw-rw-r-- root/root     11910 2009-04-21 11:44 drbd-8.3.2/drbd/connector.c
-rw-rw-r-- root/root     41155 2009-06-25 15:13 drbd-8.3.2/drbd/drbd_actlog.c
-rw-rw-r-- root/root     35585 2009-07-03 11:48 drbd-8.3.2/drbd/drbd_bitmap.c
-rw-rw-r-- root/root     77361 2009-06-25 15:13 drbd-8.3.2/drbd/drbd_int.h
-rw-rw-r-- root/root    106676 2009-06-25 15:13 drbd-8.3.2/drbd/drbd_main.c
-rw-rw-r-- root/root     66853 2009-06-25 15:13 drbd-8.3.2/drbd/drbd_nl.c
-rw-rw-r-- root/root      7485 2009-06-09 13:33 drbd-8.3.2/drbd/drbd_proc.c
-rw-rw-r-- root/root    120400 2009-06-25 15:13 drbd-8.3.2/drbd/drbd_receiver.c
-rw-rw-r-- root/root     36980 2009-06-26 11:58 drbd-8.3.2/drbd/drbd_req.c
-rw-rw-r-- root/root      9512 2009-06-09 13:33 drbd-8.3.2/drbd/drbd_req.h
-rw-rw-r-- root/root      4238 2009-06-09 13:33 drbd-8.3.2/drbd/drbd_strings.c
-rw-rw-r-- root/root     21354 2009-06-25 15:13 drbd-8.3.2/drbd/drbd_tracing.c
-rw-rw-r-- root/root      2655 2009-06-09 13:33 drbd-8.3.2/drbd/drbd_tracing.h
-rw-rw-r-- root/root     11513 2009-06-09 13:33 drbd-8.3.2/drbd/drbd_vli.h
-rw-rw-r-- root/root     40821 2009-07-03 11:46 drbd-8.3.2/drbd/drbd_worker.c
-rw-rw-r-- root/root     14925 2009-06-25 15:13 drbd-8.3.2/drbd/drbd_wrappers.h
-rw-rw-r-- root/root      4091 2009-04-21 11:44 drbd-8.3.2/drbd/linux/connector.h
drwxr-xr-x                  Creating directory: drbd-8.3.2/drbd/linux
-rw-rw-r-- root/root      9901 2009-06-15 11:02 drbd-8.3.2/drbd/linux/drbd.h
-rw-rw-r-- root/root      2861 2009-07-03 13:07 drbd-8.3.2/drbd/linux/drbd_config.h
-rw-rw-r-- root/root      3897 2009-06-09 13:33 drbd-8.3.2/drbd/linux/drbd_limits.h
-rw-rw-r-- root/root      4068 2009-06-09 13:33 drbd-8.3.2/drbd/linux/drbd_nl.h
-rw-rw-r-- root/root      2772 2009-05-26 14:49 drbd-8.3.2/drbd/linux/drbd_tag_magic.h
-rw-rw-r-- root/root        26 2009-04-21 11:44 drbd-8.3.2/drbd/linux/hardirq.h
-rw-rw-r-- root/root     12315 2009-06-09 13:33 drbd-8.3.2/drbd/linux/lru_cache.h
-rw-rw-r-- root/root       133 2009-05-26 14:49 drbd-8.3.2/drbd/linux/memcontrol.h
-rw-rw-r-- root/root       611 2009-06-09 13:33 drbd-8.3.2/drbd/linux/mutex.h
-rw-rw-r-- root/root        19 2009-06-09 13:33 drbd-8.3.2/drbd/linux/tracepoint.h
-rw-rw-r-- root/root     14751 2009-06-17 13:39 drbd-8.3.2/drbd/lru_cache.c
-rw-rw-r-- root/root      5380 2009-06-25 12:21 drbd-8.3.2/scripts/Makefile
drwxr-xr-x                  Creating directory: drbd-8.3.2/scripts
-rw-rw-r-- root/root       755 2009-06-09 13:33 drbd-8.3.2/scripts/README
-rwxrwxr-x root/root      5168 2009-06-15 14:04 drbd-8.3.2/scripts/adjust_drbd_config_h.sh
-rwxrwxr-x root/root      8047 2009-05-26 14:49 drbd-8.3.2/scripts/block-drbd
-rwxrwxr-x root/root      5813 2009-06-24 13:30 drbd-8.3.2/scripts/crm-fence-peer.sh
-rwxrwxr-x root/root      5607 2009-06-09 13:33 drbd-8.3.2/scripts/drbd
-rwxrwxr-x root/root      6250 2009-06-09 13:33 drbd-8.3.2/scripts/drbd-overview.pl
-rw-rw-r-- root/root     23765 2009-07-03 13:39 drbd-8.3.2/scripts/drbd.conf
-rw-rw-r-- root/root      3380 2008-11-24 11:43 drbd-8.3.2/scripts/drbd.gentoo
-rw-rw-r-- root/root      1420 2009-05-26 14:49 drbd-8.3.2/scripts/drbd.metadata.rhcs
-rw-rw-r-- root/root     22159 2009-07-02 14:26 drbd-8.3.2/scripts/drbd.ocf
-rw-rw-r-- root/root       649 2009-06-09 13:33 drbd-8.3.2/scripts/drbd.rules
-rwxrwxr-x root/root      3353 2009-05-26 14:49 drbd-8.3.2/scripts/drbd.sh.rhcs
-rw-rw-r-- root/root      4514 2009-05-26 14:49 drbd-8.3.2/scripts/drbdadm.bash_completion
-rwxrwxr-x root/root      2141 2009-05-26 14:49 drbd-8.3.2/scripts/drbddisk
-rw-rw-r-- root/root      1117 2009-05-26 14:49 drbd-8.3.2/scripts/drbdupper
-rwxrwxr-x root/root       334 2008-11-24 11:43 drbd-8.3.2/scripts/get_uts_release.sh
-rwxrwxr-x root/root      3526 2009-05-26 14:49 drbd-8.3.2/scripts/notify.sh
-rwxrwxr-x root/root      3015 2008-11-24 11:43 drbd-8.3.2/scripts/outdate-peer.sh
```

```
-rw-rw-r-- root/root      5414 2009-06-09 13:33 drbd-8.3.2/scripts/patch-kernel
-rw-rw-r-- root/root      2732 2009-05-26 14:49 drbd-8.3.2/scripts/pretty-proc-drbd.sh
-rwxrwxr-x root/root      2099 2009-05-26 14:49 drbd-8.3.2/scripts/snapshot-resync-target-lvm.sh
lrwxrwxrwx root/root         0 2009-08-31 13:06 drbd-8.3.2/scripts/unsnapshot-resync-target-lvm.sh ->
   snapshot-resync-target-lvm.sh
-rw-rw-r-- root/root      3561 2009-06-26 14:42 drbd-8.3.2/user/Makefile
drwxr-xr-x                  Creating directory: drbd-8.3.2/user
-rw-rw-r-- root/root      5098 2009-04-21 11:44 drbd-8.3.2/user/drbd_endian.h
-rw-rw-r-- root/root      7282 2009-06-24 13:30 drbd-8.3.2/user/drbdadm.h
-rw-rw-r-- root/root      6590 2009-06-24 13:30 drbd-8.3.2/user/drbdadm_adjust.c
-rw-rw-r-- root/root     79859 2009-06-29 11:43 drbd-8.3.2/user/drbdadm_main.c
-rw-rw-r-- root/root      4314 2009-05-26 14:49 drbd-8.3.2/user/drbdadm_minor_table.c
-rw-rw-r-- root/root     39096 2009-07-02 14:26 drbd-8.3.2/user/drbdadm_parser.c
-rw-rw-r-- root/root      2612 2009-06-09 13:33 drbd-8.3.2/user/drbdadm_parser.h
-rw-rw-r-- root/root      7531 2009-06-10 13:36 drbd-8.3.2/user/drbdadm_scanner.fl
-rw-rw-r-- root/root     17911 2009-06-17 15:58 drbd-8.3.2/user/drbdadm_usage_cnt.c
-rw-rw-r-- root/root     75570 2009-06-10 13:36 drbd-8.3.2/user/drbdmeta.c
-rw-rw-r-- root/root       474 2009-04-21 11:44 drbd-8.3.2/user/drbdmeta_parser.h
-rw-rw-r-- root/root      2306 2009-06-10 13:36 drbd-8.3.2/user/drbdmeta_scanner.fl
-rw-rw-r-- root/root     70962 2009-06-29 11:31 drbd-8.3.2/user/drbdsetup.c
-rw-rw-r-- root/root     14240 2009-06-15 16:44 drbd-8.3.2/user/drbdtool_common.c
-rw-rw-r-- root/root      2210 2009-06-10 13:36 drbd-8.3.2/user/drbdtool_common.h
-rw-rw-r-- root/root      1783 2009-06-18 17:28 drbd-8.3.2/user/unaligned.h
-rw-rw-r-- root/root     33221 2009-07-03 15:35 drbd-8.3.2/documentation/drbdsetup.8
-rw-rw-r-- root/root      7652 2009-07-03 15:35 drbd-8.3.2/documentation/drbdadm.8
-rw-rw-r-- root/root      1825 2009-07-03 15:35 drbd-8.3.2/documentation/drbd.8
-rw-rw-r-- root/root      3493 2009-07-03 15:35 drbd-8.3.2/documentation/drbdmeta.8
-rw-rw-r-- root/root     33933 2009-07-03 15:35 drbd-8.3.2/documentation/drbd.conf.5
-rw-rw-r-- root/root      1687 2009-07-03 15:35 drbd-8.3.2/documentation/drbddisk.8
lrwxrwxrwx root/root         0 2009-08-31 13:09 drbd-8.3.2/drbd_config.h -> drbd/linux/drbd_config.h
-rw-r--r-- root/root       228 2009-08-31 13:09 drbd-8.3.2/drbd/drbd_buildtag.c
-rw-rw-r-- root/root      3157 2009-07-03 15:35 drbd-8.3.2/.filelist
+ STATUS=0
+ '[' 0 -ne 0 ']'
+ cd drbd-8.3.2
+ /bin/chmod -Rf a+rX,u+w,g-w,o-w .
+ test -d /lib/modules/2.6.29.6-217.2.16.fc11.x86_64/build/.
++ scripts/get_uts_release.sh
+ test 2.6.29.6-217.2.16.fc11.x86_64 = 2.6.29.6-217.2.16.fc11.x86_64
+ exit 0
Executing(%build): /bin/sh -e /var/tmp/rpm-tmp.oZdvus
+ umask 022
+ cd /root/drbd-8.3.2/dist/BUILD
+ cd drbd-8.3.2
+ echo kernelversion=2.6.29.6-217.2.16.fc11.x86_64
kernelversion=2.6.29.6-217.2.16.fc11.x86_64
+ echo 'kversion=%{kversion}'
kversion=%{kversion}
+ echo krelver=2.6.29.6_217.2.16.fc11.x86_64
krelver=2.6.29.6_217.2.16.fc11.x86_64
+ '[' -n /root/drbd-8.3.2/dist/install -a /root/drbd-8.3.2/dist/install '!=' / ']'
+ rm -rf /root/drbd-8.3.2/dist/install
+ mkdir -p /root/drbd-8.3.2/dist/install
+ make clean
make[1]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2'
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/user'
rm -f drbdadm_scanner.c drbdmeta_scanner.c
rm -f drbdsetup drbdadm drbdmeta drbdmeta_unfinished_rewrite *.o
```

```
rm -f drbd_buildtag.c drbd_strings.c
rm -f *~
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/user'
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/scripts'
rm -f *~
rm -f datadisk
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/scripts'
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/documentation'
To clean the documentation: make doc-clean
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/documentation'
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd'
rm -rf .tmp_versions
rm -f *.[oas] *.ko .*.cmd .*.d .*.tmp *.mod.c .*.flags .depend .kernel*
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd'
rm -f *~
rm -rf dist
make[1]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2'
+ '[' -e /etc/redhat-release ']'
+ make all doc PREFIX=/root/drbd-8.3.2/dist/install/ MANDIR=/usr/share/man KDIR=/lib/modules/
   2.6.29.6-217.2.16.fc11.x86_64/build LOCALVERSION=
make[1]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2'
make -C drbd drbd_buildtag.c
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd'
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd'
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/user'
flex -s -odrbdadm_scanner.c drbdadm_scanner.fl
gcc -O1 -c -W -Wall -I../drbd   -c -o drbdadm_scanner.o drbdadm_scanner.c
gcc -O1 -c -W -Wall -I../drbd   -c -o drbdadm_parser.o drbdadm_parser.c
gcc -O1 -c -W -Wall -I../drbd   -c -o drbdadm_main.o drbdadm_main.c
drbdadm_main.c: In function ‘sanity_check_abs_cmd’:
drbdadm_main.c:2440: warning: suggest parentheses around operand of ‘!’ or change ‘&’ to ‘&&’ or ‘!’ to ‘~’
gcc -O1 -c -W -Wall -I../drbd   -c -o drbdadm_adjust.o drbdadm_adjust.c
gcc -O1 -c -W -Wall -I../drbd   -c -o drbdtool_common.o drbdtool_common.c
gcc -O1 -c -W -Wall -I../drbd   -c -o drbdadm_usage_cnt.o drbdadm_usage_cnt.c
cp ../drbd/drbd_buildtag.c drbd_buildtag.c
gcc -O1 -c -W -Wall -I../drbd   -c -o drbd_buildtag.o drbd_buildtag.c
gcc -O1 -c -W -Wall -I../drbd   -c -o drbdadm_minor_table.o drbdadm_minor_table.c
gcc -o drbdadm drbdadm_scanner.o drbdadm_parser.o drbdadm_main.o drbdadm_adjust.o drbdtool_common.o
   drbdadm_usage_cnt.o drbd_buildtag.o drbdadm_minor_table.o
gcc -O1 -c -W -Wall -I../drbd   -c -o drbdmeta.o drbdmeta.c
flex -s -odrbdmeta_scanner.c drbdmeta_scanner.fl
gcc -O1 -c -W -Wall -I../drbd   -c -o drbdmeta_scanner.o drbdmeta_scanner.c
gcc -o drbdmeta drbdmeta.o drbdmeta_scanner.o drbdtool_common.o drbd_buildtag.o
gcc -O1 -c -W -Wall -I../drbd -I/lib/modules/2.6.29.6-217.2.16.fc11.x86_64/build/include   -c -o drbdsetup.o
   drbdsetup.c
cp ../drbd/drbd_strings.c drbd_strings.c
gcc -O1 -c -W -Wall -I../drbd -I/lib/modules/2.6.29.6-217.2.16.fc11.x86_64/build/include   -c -o
   drbd_strings.o drbd_strings.c
gcc -o drbdsetup drbdsetup.o drbdtool_common.o drbd_buildtag.o drbd_strings.o
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/user'
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/scripts'
make[2]: Nothing to be done for `all'.
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/scripts'
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/documentation'
To (re)make the documentation: make doc
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/documentation'

   Userland tools build was successful.
```

```
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd'

    Calling toplevel makefile of kernel source tree, which I believe is in
    KDIR=/lib/modules/2.6.29.6-217.2.16.fc11.x86_64/build

test -f ../scripts/adjust_drbd_config_h.sh && \
    KDIR=/lib/modules/2.6.29.6-217.2.16.fc11.x86_64/build O= /bin/bash ../scripts/adjust_drbd_config_h.sh
/lib/modules/2.6.29.6-217.2.16.fc11.x86_64/build ~/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd
~/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd

  Using unmodified drbd_config.h

make -C /lib/modules/2.6.29.6-217.2.16.fc11.x86_64/build SUBDIRS=/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd
   modules
make[3]: Entering directory `/usr/src/kernels/2.6.29.6-217.2.16.fc11.x86_64'
  CC [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd_buildtag.o
  CC [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd_bitmap.o
  CC [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd_proc.o
  CC [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd_worker.o
  CC [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd_receiver.o
  CC [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd_req.o
  CC [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd_actlog.o
  CC [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/lru_cache.o
  CC [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd_main.o
  CC [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd_strings.o
  CC [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd_nl.o
  LD [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC      /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd.mod.o
  LD [M]  /root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd/drbd.ko
make[3]: Leaving directory `/usr/src/kernels/2.6.29.6-217.2.16.fc11.x86_64'
mv .drbd_kernelrelease.new .drbd_kernelrelease
Memorizing module configuration ... done.
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd'

   Module build was successful.
make -C documentation doc
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/documentation'
make[2]: Nothing to be done for `doc'.
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/documentation'
make[1]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2'
+ exit 0
Executing(%install): /bin/sh -e /var/tmp/rpm-tmp.LIk02d
+ umask 022
+ cd /root/drbd-8.3.2/dist/BUILD
+ cd drbd-8.3.2
+ make install PREFIX=/root/drbd-8.3.2/dist/install/ MANDIR=/usr/share/man
make[1]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2'
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/user'
install -d /root/drbd-8.3.2/dist/install//sbin/
if getent group haclient > /dev/null 2> /dev/null ; then \
          install -g haclient -m 4750 drbdsetup /root/drbd-8.3.2/dist/install//sbin/ ;     \
          install -g haclient -m 4750 drbdmeta /root/drbd-8.3.2/dist/install//sbin/ ;      \
          install -m 755 drbdadm /root/drbd-8.3.2/dist/install//sbin/ ;             \
   else                                                 \
          install -m 755 drbdsetup /root/drbd-8.3.2/dist/install//sbin/ ;           \
          install -m 755 drbdmeta /root/drbd-8.3.2/dist/install//sbin/ ;            \
```

```
           install -m 755 drbdadm /root/drbd-8.3.2/dist/install//sbin/ ;             \
    fi
mkdir -p /root/drbd-8.3.2/dist/install//var/lib/drbd
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/user'
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/scripts'
mkdir -p /root/drbd-8.3.2/dist/install//etc/ha.d/resource.d
install -d /root/drbd-8.3.2/dist/install//etc/rc.d/init.d/
install -m 755 drbd /root/drbd-8.3.2/dist/install//etc/rc.d/init.d/drbd
install -m 755 drbddisk /root/drbd-8.3.2/dist/install//etc/ha.d/resource.d
install -m 755 drbdupper /root/drbd-8.3.2/dist/install//etc/ha.d/resource.d
mkdir -p /root/drbd-8.3.2/dist/install//usr/lib/ocf/resource.d/linbit
install -m 755 drbd.ocf /root/drbd-8.3.2/dist/install//usr/lib/ocf/resource.d/linbit/drbd
install -d /root/drbd-8.3.2/dist/install//usr/lib/drbd
install -m 755 outdate-peer.sh /root/drbd-8.3.2/dist/install//usr/lib/drbd
install -m 755 crm-fence-peer.sh /root/drbd-8.3.2/dist/install//usr/lib/drbd
install -m 755 snapshot-resync-target-lvm.sh /root/drbd-8.3.2/dist/install//usr/lib/drbd
install -m 755 notify.sh /root/drbd-8.3.2/dist/install//usr/lib/drbd
( set -e ; cd /root/drbd-8.3.2/dist/install//usr/lib/drbd ;\
     ln -sf crm-fence-peer.sh crm-unfence-peer.sh ;\
     ln -sf snapshot-resync-target-lvm.sh unsnapshot-resync-target-lvm.sh ;\
     ln -sf notify.sh notify-split-brain.sh ;\
     ln -sf notify.sh notify-io-error.sh ;\
     ln -sf notify.sh notify-pri-on-incon-degr.sh ;\
     ln -sf notify.sh notify-pri-lost.sh ;\
     ln -sf notify.sh notify-pri-lost-after-sb.sh ;\
     ln -sf notify.sh notify-emergency-reboot.sh ;\
     ln -sf notify.sh notify-emergency-shutdown.sh ;\
     ln -sf notify.sh notify-out-of-sync.sh; )
mkdir -p /root/drbd-8.3.2/dist/install//etc/xen/scripts
install -m 755 block-drbd /root/drbd-8.3.2/dist/install//etc/xen/scripts
mkdir -p /root/drbd-8.3.2/dist/install//etc/udev/rules.d
mkdir -p /root/drbd-8.3.2/dist/install//etc/bash_completion.d
install -m 644 drbdadm.bash_completion /root/drbd-8.3.2/dist/install//etc/bash_completion.d/drbdadm
install -d /root/drbd-8.3.2/dist/install//usr/sbin
# strip .pl on install,
# at some point we might reimplement similar functionality in non-perl
install -m 755 drbd-overview.pl /root/drbd-8.3.2/dist/install//usr/sbin/drbd-overview
mkdir -p /root/drbd-8.3.2/dist/install//usr/share/cluster
install -m 755 drbd.sh.rhcs /root/drbd-8.3.2/dist/install//usr/share/cluster/drbd.sh
install -m 644 drbd.metadata.rhcs /root/drbd-8.3.2/dist/install//usr/share/cluster/drbd.metadata

Don't forget to run update-rc.d or chkconfig
disable=.disabled ; \
   v=$(udevadm version 2>/dev/null) || \
   v=$(rpm -q --qf '%{VERSION}' udev 2>/dev/null) || \
   v=$(dpkg-query -W -f '${Version}' udev 2>/dev/null); \
   v=${v##*.} ; v=${v%%-*} ; v=${v#0} ; \
   if [ -n "$DRBD_ENABLE_UDEV" ] ; then \
          disable="" ; \
   else case "$v" in \
   "") :;; *[!0-9]*) :;; *) \
          if [ "$v" -ge 85 ]; then \
                disable=""; \
          fi ;; \
   esac; fi ; \
   install -m 644 drbd.rules /root/drbd-8.3.2/dist/install//etc/udev/rules.d/65-drbd.rules$disable
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/scripts'
```

```
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/documentation'
set -e; for f in drbdsetup.8 drbd.conf.5 drbd.8 drbdadm.8 drbddisk.8 drbdmeta.8 ; do \
          s=${f##*.}; \
          install -v -D -m 644 $f /root/drbd-8.3.2/dist/install//usr/share/man/man$s/$f ; \
   done
install: creating directory `/root/drbd-8.3.2/dist/install//usr/share/man'
install: creating directory `/root/drbd-8.3.2/dist/install//usr/share/man/man8'
`drbdsetup.8' -> `/root/drbd-8.3.2/dist/install//usr/share/man/man8/drbdsetup.8'
install: creating directory `/root/drbd-8.3.2/dist/install//usr/share/man/man5'
`drbd.conf.5' -> `/root/drbd-8.3.2/dist/install//usr/share/man/man5/drbd.conf.5'
`drbd.8' -> `/root/drbd-8.3.2/dist/install//usr/share/man/man8/drbd.8'
`drbdadm.8' -> `/root/drbd-8.3.2/dist/install//usr/share/man/man8/drbdadm.8'
`drbddisk.8' -> `/root/drbd-8.3.2/dist/install//usr/share/man/man8/drbddisk.8'
`drbdmeta.8' -> `/root/drbd-8.3.2/dist/install//usr/share/man/man8/drbdmeta.8'
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/documentation'
make[2]: Entering directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd'
install -d /root/drbd-8.3.2/dist/install//lib/modules/2.6.29.6-217.2.16.fc11.x86_64/kernel/drivers/block
install -m 644 drbd.ko /root/drbd-8.3.2/dist/install//lib/modules/2.6.29.6-217.2.16.fc11.x86_64/kernel/
   drivers/block
make[2]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/drbd'
make[1]: Leaving directory `/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2'
+ cd drbd
+ mv .kernel.config.gz k-config-2.6.29.6-217.2.16.fc11.x86_64.gz
+ FILELIST=/root/drbd-8.3.2/dist/BUILD/drbd-8.3.2/file.list
+ cd /root/drbd-8.3.2/dist/install
+ find etc/ -name drbd -printf '/%p\n'
+ test -e sbin/rcdrbd
+ test -e etc/bash_completion.d/drbdadm
+ echo /etc/bash_completion.d/drbdadm
+ test -e etc/bash_completion.d/drbdadm.sh
+ test -e usr/share/cluster/drbd.sh
+ echo /usr/share/cluster/drbd.sh
+ test -e usr/share/cluster/drbd.metadata
+ echo /usr/share/cluster/drbd.metadata
+ cat
+ /usr/lib/rpm/brp-compress
+ /usr/lib/rpm/brp-strip
+ /usr/lib/rpm/brp-strip-static-archive
+ /usr/lib/rpm/brp-strip-comment-note
Processing files: drbd-8.3.2-3.x86_64
Executing(%doc): /bin/sh -e /var/tmp/rpm-tmp.ZP27D3
+ umask 022
+ cd /root/drbd-8.3.2/dist/BUILD
+ cd drbd-8.3.2
+ DOCDIR=/root/drbd-8.3.2/dist/install/usr/share/doc/drbd-8.3.2
+ export DOCDIR
+ rm -rf /root/drbd-8.3.2/dist/install/usr/share/doc/drbd-8.3.2
+ /bin/mkdir -p /root/drbd-8.3.2/dist/install/usr/share/doc/drbd-8.3.2
+ cp -pr scripts/drbd.conf /root/drbd-8.3.2/dist/install/usr/share/doc/drbd-8.3.2
+ cp -pr COPYING /root/drbd-8.3.2/dist/install/usr/share/doc/drbd-8.3.2
+ cp -pr README /root/drbd-8.3.2/dist/install/usr/share/doc/drbd-8.3.2
+ cp -pr file.list /root/drbd-8.3.2/dist/install/usr/share/doc/drbd-8.3.2
+ exit 0
Provides: config(drbd) = 8.3.2-3
Requires(interp): /bin/sh /bin/sh
Requires(rpmlib): rpmlib(CompressedFileNames) <= 3.0.4-1 rpmlib(PayloadFilesHavePrefix) <= 4.0-1
Requires(post): /bin/sh
Requires(preun): /bin/sh
```

```
Requires: /bin/bash /usr/bin/perl libc.so.6()(64bit) libc.so.6(GLIBC_2.2.5)(64bit) libc.so.6(GLIBC_2.3)
   (64bit) libc.so.6(GLIBC_2.7)(64bit) perl(strict) perl(warnings) rtld(GNU_HASH)
Processing files: drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64
Executing(%doc): /bin/sh -e /var/tmp/rpm-tmp.x2DVwT
+ umask 022
+ cd /root/drbd-8.3.2/dist/BUILD
+ cd drbd-8.3.2
+ DOCDIR=/root/drbd-8.3.2/dist/install/usr/share/doc/drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2
+ export DOCDIR
+ rm -rf /root/drbd-8.3.2/dist/install/usr/share/doc/drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2
+ /bin/mkdir -p /root/drbd-8.3.2/dist/install/usr/share/doc/drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2
+ cp -pr drbd/k-config-2.6.29.6-217.2.16.fc11.x86_64.gz /root/drbd-8.3.2/dist/install/usr/share/doc/drbd-
   km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2
+ exit 0
Requires(interp): /bin/sh /bin/sh
Requires(rpmlib): rpmlib(CompressedFileNames) <= 3.0.4-1 rpmlib(PayloadFilesHavePrefix) <= 4.0-1
Requires(post): /bin/sh
Requires(postun): /bin/sh
Conflicts: km_drbd drbd-kmp <= 8.3.2_3 drbd-kmod <= 8.3.2_3
Checking for unpackaged file(s): /usr/lib/rpm/check-files /root/drbd-8.3.2/dist/install
Wrote: /root/drbd-8.3.2/dist/RPMS/x86_64/drbd-8.3.2-3.x86_64.rpm
Wrote: /root/drbd-8.3.2/dist/RPMS/x86_64/drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64.rpm
Executing(%clean): /bin/sh -e /var/tmp/rpm-tmp.clqiey
+ umask 022
+ cd /root/drbd-8.3.2/dist/BUILD
+ cd drbd-8.3.2
+ '[' -n /root/drbd-8.3.2/dist/install -a /root/drbd-8.3.2/dist/install '!=' / ']'
+ rm -rf /root/drbd-8.3.2/dist/install
+ exit 0
You have now:
-rw-r--r-- 1 root root  220953 2009-08-31 13:09 dist/RPMS/x86_64/drbd-8.3.2-3.x86_64.rpm
-rw-r--r-- 1 root root 1145718 2009-08-31 13:09 dist/RPMS/x86_64/drbd-
   km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64.rpm
[root@test1 drbd-8.3.2]#
```

Install the DRBD Packages

The completed build process will store the result in the dist/RPMS/x86_64/ subdirectory and all that is required now is to install them with YUM.

yum localinstall -y --nogpgcheck dist/RPMS/x86_64/drbd-*.rpm

```
[root@test1 drbd-8.3.2]# yum localinstall -y --nogpgcheck dist/RPMS/x86_64/drbd-*.rpm
Loaded plugins: refresh-packagekit, versionlock
Setting up Local Package Process
Examining dist/RPMS/x86_64/drbd-8.3.2-3.x86_64.rpm: drbd-8.3.2-3.x86_64
Marking dist/RPMS/x86_64/drbd-8.3.2-3.x86_64.rpm to be installed
Reading version lock configuration
Examining dist/RPMS/x86_64/drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64.rpm: drbd-
   km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64
Marking dist/RPMS/x86_64/drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package drbd.x86_64 0:8.3.2-3 set to be updated
---> Package drbd-km-2.6.29.6_217.2.16.fc11.x86_64.x86_64 0:8.3.2-3 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package
      Arch   Version  Repository                                          Size
================================================================================
Installing:
 drbd x86_64 8.3.2-3  /drbd-8.3.2-3.x86_64                               422 k
 drbd-km-2.6.29.6_217.2.16.fc11.x86_64
      x86_64 8.3.2-3  /drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64 3.7 M

Transaction Summary
================================================================================
Install      2 Package(s)
Update       0 Package(s)
Remove       0 Package(s)

Total size: 4.1 M
Downloading Packages:
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing     : drbd-8.3.2-3.x86_64                                     1/2
  Installing     : drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64    2/2

Installed:
  drbd.x86_64 0:8.3.2-3   drbd-km-2.6.29.6_217.2.16.fc11.x86_64.x86_64 0:8.3.2-3

Complete!
```

By default DRBD configures itself to start when the machine is powered on, however since we want the cluster to manage it, we will need to disable this behavior:

chkconfig --del drbd

We could rebuild the drbd package on *test2*, however if they share the same architecture (x86_64 in this case) we can reuse the ones we built for *test1*. Assuming this is the case for you, copy them to *test2* and install:

scp dist/RPMS/x86_64/drbd-*.rpm test2:

ssh test2 -- yum localinstall -y --nogpgcheck drbd-*.rpm

chkconfig --del drbd

```
[root@test1 drbd-8.3.2]# scp dist/RPMS/x86_64/drbd-*.rpm test2:
drbd-8.3.2-3.x86_64.rpm
   100%  216KB 215.8KB/s   00:01
drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64.rpm
   100% 1119KB   1.1MB/s   00:00
[root@test1 drbd-8.3.2]# ssh test2 -- yum localinstall -y --nogpgcheck drbd-*.rpm
Loaded plugins: refresh-packagekit, versionlock
Setting up Local Package Process
Examining drbd-8.3.2-3.x86_64.rpm: drbd-8.3.2-3.x86_64
Marking drbd-8.3.2-3.x86_64.rpm to be installed
Reading version lock configuration
Examining drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64.rpm: drbd-
   km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64
Marking drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package drbd.x86_64 0:8.3.2-3 set to be updated
---> Package drbd-km-2.6.29.6_217.2.16.fc11.x86_64.x86_64 0:8.3.2-3 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package
   Arch   Version   Repository                                          Size
================================================================================
Installing:
 drbd
   x86_64 8.3.2-3   /drbd-8.3.2-3.x86_64                               422 k
 drbd-km-2.6.29.6_217.2.16.fc11.x86_64
   x86_64 8.3.2-3   /drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64 3.7 M

Transaction Summary
================================================================================
Install      2 Package(s)
Update       0 Package(s)
Remove       0 Package(s)

Total size: 4.1 M
Downloading Packages:
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing     : drbd-8.3.2-3.x86_64                                     1/2
  Installing     : drbd-km-2.6.29.6_217.2.16.fc11.x86_64-8.3.2-3.x86_64    2/2
```

```
Installed:
  drbd.x86_64 0:8.3.2-3  drbd-km-2.6.29.6_217.2.16.fc11.x86_64.x86_64 0:8.3.2-3

Complete!
[root@test1 drbd-8.3.2]#
```

Configure DRBD

Before we configure DRBD, we need to set aside some disk for it to use.

Optional Horrible Hack

By default, Fedora 11 will use all the available hard-disk for the operating system, with the exception of a small swap partition.

If you arrived at this point without any space for a new LVM partition, you can cheat and steal some from the swap device. This is less than an ideal solution, but it is significantly simpler than resizing a live root filesystem.

First we identify the existing swap partition

lvs

```
[root@test1 drbd-8.3.2]# lvs
  LV      VG       Attr   LSize   Origin Snap%  Move Log Copy%  Convert
  lv_root vg_test1 -wi-ao   7.30G
  lv_swap vg_test1 -wi-ao 512.00M
```

Then we deactivate it, remove it and recreate a slightly smaller version

swapoff -a

lvremove -f vg_test1/lv_swap

lvcreate -n lv_swap -L 500m vg_test1

mkswap /dev/mapper/vg_test1-lv_swap

swapon -a

lvs

```
[root@test1 ~]# swapoff -a
[root@test1 drbd-8.3.2]# lvremove -f vg_test1/lv_swap
  Logical volume "lv_swap" successfully removed
[root@test1 drbd-8.3.2]# lvcreate -n lv_swap -L 500m vg_test1
  Logical volume "lv_swap" created
[root@test1 drbd-8.3.2]# mkswap /dev/mapper/vg_test1-lv_swap
Setting up swapspace version 1, size = 511996 KiB
no label, UUID=aaddc16a-b307-46b8-9efe-93cd8bbfc08d
[root@test1 drbd-8.3.2]# swapon -a
[root@test1 drbd-8.3.2]# lvs
  LV      VG       Attr   LSize   Origin Snap%  Move Log Copy%  Convert
  lv_root vg_test1 -wi-ao   7.30G
  lv_swap vg_test1 -wi-ao 500.00M
```

Now we have room to create a small (10Mb) partition for DRBD.

Now repeat the process on test2.

lvs

swapoff -a

lvremove -f vg_test2/lv_swap

lvcreate -n lv_swap -L 500m vg_test2

mkswap /dev/mapper/vg_test2-lv_swap

swapon -a

lvs

```
[root@test1 drbd-8.3.2]# lvs
  LV      VG       Attr   LSize   Origin Snap%  Move Log Copy%  Convert
  lv_root vg_test2 -wi-ao   7.30G
  lv_swap vg_test2 -wi-ao 512.00M
[root@test2 ~]# swapoff -a
[root@test2 ~]# lvremove -f vg_test2/lv_swap
  Logical volume "lv_swap" successfully removed
[root@test2 ~]# lvcreate -n lv_swap -L 500m vg_test2
  Logical volume "lv_swap" created
[root@test2 ~]# mkswap /dev/mapper/vg_test2-lv_swap
Setting up swapspace version 1, size = 511996 KiB
no label, UUID=aaddc16a-b307-46b8-9efe-93cd8bbfc08d
[root@test2 ~]# swapon -a
[root@test2 ~]# lvs
  LV      VG       Attr   LSize   Origin Snap%  Move Log Copy%  Convert
  lv_root vg_test2 -wi-ao   7.30G
  lv_swap vg_test2 -wi-ao 500.00M
```

Create A Partition for DRBD

If you have more than 10Mb free, feel free to use it. For this guide however, 10Mb is enough for a single html file.

lvcreate -n drbd-demo -L 10m vg_test1

lvs

```
[root@test1 drbd-8.3.2]# lvcreate -n drbd-demo -L 10m vg_test1
  Rounding up size to full physical extent 12.00 MB
  Logical volume "drbd-demo" created
[root@test1 drbd-8.3.2]# lvs
  LV        VG       Attr   LSize   Origin Snap%  Move Log Copy%  Convert
  drbd-demo vg_test1 -wi-a-  12.00M
  lv_root   vg_test1 -wi-ao   7.30G
  lv_swap   vg_test1 -wi-ao 500.00M
```

Repeat this on the second node, be sure to use the same size partition.

lvs

lvcreate -n drbd-demo -L 10m vg_test2

lvs

```
[root@test2 ~]# lvs
  LV      VG       Attr   LSize   Origin Snap%  Move Log Copy%  Convert
  lv_root vg_test2 -wi-ao   7.30G
  lv_swap vg_test2 -wi-ao 500.00M
[root@test2 ~]# lvcreate -n drbd-demo -L 10m vg_test2
```

```
  Rounding up size to full physical extent 12.00 MB
  Logical volume "drbd-demo" created
[root@test2 ~]# lvs
  LV        VG       Attr   LSize   Origin Snap%  Move Log Copy%  Convert
  drbd-demo vg_test2 -wi-a-  12.00M
  lv_root   vg_test2 -wi-ao   7.30G
  lv_swap   vg_test2 -wi-ao 500.00M
```

Write the DRBD Config

There is no series of commands for build a DRBD configuration, so simply copy the configuration below to */etc/drbd.conf*

Detailed information on the directives used in this configuration (and other alternatives) is available from http://www.drbd.org/users-guide/ch-configure.html

Be sure to use the names and addresses of *your* nodes if they differ from the ones used in this guide.

```
global {
  usage-count yes;
}
common {
  protocol C;
}
resource wwwdata {
  meta-disk internal;
  device    /dev/drbd1;
  syncer {
    verify-alg sha1;
  }
  net {
    allow-two-primaries;
  }
  on test1 {
    disk      /dev/dev/mapper/vg_test1-drbd--demo;
    address   192.168.9.41:7789;
  }
  on test2 {
    disk      /dev/dev/mapper/vg_test2-drbd--demo;
    address   192.168.9.42:7789;
  }
}
```

Initialize and Load DRBD

With the configuration in place, we can now perform the DRBD initialization

drbdadm create-md wwwdata

```
[root@test1 drbd-8.3.2]# drbdadm create-md wwwdata
md_offset 12578816
al_offset 12546048
bm_offset 12541952

Found some data
 ==> This might destroy existing data! <==

Do you want to proceed?
[need to type 'yes' to confirm] yes

Writing meta data...
initializing activity log
NOT initialized bitmap
New drbd meta data block successfully created.
success
```

Now load the DRBD kernel module and confirm that everything is sane

modprobe drbd

drbdadm up wwwdata

cat /proc/drbd

```
[root@test1 drbd-8.3.2]# modprobe drbd
[root@test1 drbd-8.3.2]# drbdadm up wwwdata
[root@test1 drbd-8.3.2]# cat /proc/drbd
version: 8.3.2 (api:88/proto:86-90)
GIT-hash: dd7985327f146f33b86d4bff5ca8c94234ce840e build by root@test1.clusterlabs.org, 2009-08-31 13:09:34

 1: cs:WFConnection ro:Secondary/Unknown ds:Inconsistent/DUnknown C r----
    ns:0 nr:0 dw:0 dr:0 al:0 bm:0 lo:0 pe:0 ua:0 ap:0 ep:1 wo:b oos:12248
[root@test1 drbd-8.3.2]#
```

Repeat on the second node

drbdadm --force create-md wwwdata

modprobe drbd

drbdadm up wwwdata

cat /proc/drbd

```
[root@test2 ~]# drbdadm --force create-md wwwdata
Writing meta data...
initializing activity log
NOT initialized bitmap
New drbd meta data block successfully created.
success
[root@test2 ~]# modprobe drbd
WARNING: Deprecated config file /etc/modprobe.conf, all config files belong into /etc/modprobe.d/.
[root@test2 ~]# drbdadm up wwwdata
[root@test2 ~]# cat /proc/drbd
version: 8.3.2 (api:88/proto:86-90)
GIT-hash: dd7985327f146f33b86d4bff5ca8c94234ce840e build by root@test1.clusterlabs.org, 2009-08-31 13:09:34

 1: cs:Connected ro:Secondary/Secondary ds:Inconsistent/Inconsistent C r----
    ns:0 nr:0 dw:0 dr:0 al:0 bm:0 lo:0 pe:0 ua:0 ap:0 ep:1 wo:b oos:12248
```

Now we need to tell DRBD which set of data to use. Since both sides contain garbage, we can run the following on test1:

drbdadm -- --overwrite-data-of-peer primary wwwdata

cat /proc/drbd

```
[root@test1 ~]# drbdadm -- --overwrite-data-of-peer primary wwwdata
[root@test1 ~]# cat /proc/drbd
version: 8.3.2 (api:88/proto:86-90)
GIT-hash: dd7985327f146f33b86d4bff5ca8c94234ce840e build by root@test1.clusterlabs.org, 2009-08-31 13:09:34

 1: cs:SyncSource ro:Primary/Secondary ds:UpToDate/Inconsistent C r----
    ns:2184 nr:0 dw:0 dr:2472 al:0 bm:0 lo:0 pe:0 ua:0 ap:0 ep:1 wo:b oos:10064
    [=====>..............] sync'ed: 33.4% (10064/12248)K
    finish: 0:00:37 speed: 240 (240) K/sec
[root@test1 ~]# cat /proc/drbd
version: 8.3.2 (api:88/proto:86-90)
GIT-hash: dd7985327f146f33b86d4bff5ca8c94234ce840e build by root@test1.clusterlabs.org, 2009-08-31 13:09:34

 1: cs:Connected ro:Primary/Secondary ds:UpToDate/UpToDate C r----
    ns:12248 nr:0 dw:0 dr:12536 al:0 bm:1 lo:0 pe:0 ua:0 ap:0 ep:1 wo:b oos:0
```

test1 is now in the Primary state which allows it to be written to. Which means its a good point at which to create a filesystem and populate it with some data to serve up via our WebSite resource.

Populate DRBD with Data

mkfs.ext4 /dev/drbd1

```
[root@test1 ~]# mkfs.ext4 /dev/drbd1
mke2fs 1.41.4 (27-Jan-2009)
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
3072 inodes, 12248 blocks
612 blocks (5.00%) reserved for the super user
First data block=1
Maximum filesystem blocks=12582912
2 block groups
8192 blocks per group, 8192 fragments per group
1536 inodes per group
Superblock backups stored on blocks:
    8193

Writing inode tables: done
Creating journal (1024 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 26 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```

Now mount the newly created filesystem so we can create our index file

mount /dev/drbd1 /mnt/

cat <<-END >/mnt/index.html

<html>

<body>My Test Site - drbd</body>

</html>

END

umount /dev/drbd1

```
[root@test1 ~]# mount /dev/drbd1 /mnt/
[root@test1 ~]# cat <<-END >/mnt/index.html
> <html>
> <body>My Test Site - drbd</body>
> </html>
> END
[root@test1 ~]# umount /dev/drbd1
```

And finally, confirm the data is in sync between the two nodes

drbdadm verify wwwdata

echo $?

```
[root@test1 ~]# drbdadm verify wwwdata
[root@test1 ~]# echo $?
0
```

Configure the Cluster for DRBD

One handy feature of the crm shell is that you can use it in interactive mode to make several changes atomically.

First we launch the shell. The prompt will change to indicate you're in interactive mode.

crm

```
[root@test1 ~]# crm
cib crm(live)#
```

Next we must create a working copy or the current configuration. This is where all our changes will go. The cluster will not see any of them until we say its ok. Notice again how the prompt changes, this time to indicate that we're no longer looking at the live cluster.

cib new drbd

```
cib crm(live)# cib new drbd
INFO: drbd shadow CIB created
crm(drbd)#
```

Now we can create our DRBD clone and display the revised configuration.

configure primitive wwwdrbd ocf:linbit:drbd params drbd_resource=wwwdata op monitor interval=60s

configure ms WebData wwwdrbd meta master-max=1 master-node-max=1 \
clone-max=2 clone-node-max=1 notify=true

configure show

```
crm(drbd)# configure primitive ocf:linbit:drbd WebData params drbd_resource=wwwdata op monitor interval=60s
crm(drbd)# configure ms WebDataClone WebData meta master-max=1 master-node-max=1 \
   clone-max=2 clone-node-max=1 notify=true
crm(drbd)# configure show
node test1
node test2
primitive WebData ocf:linbit:drbd \
   params drbd_resource="wwwdata" \
   op monitor interval="60s"
primitive WebSite ocf:heartbeat:apache \
   params configfile="/etc/httpd/conf/httpd.conf" \
   op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
   params ip="192.168.9.101" cidr_netmask="32" \
   op monitor interval="30s"
ms WebDataClone WebData \
   meta master-max="1" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
location prefer-test1 WebSite 50: test1
colocation website-with-ip inf: WebSite ClusterIP
order apache-after-ip inf: ClusterIP WebSite
property $id="cib-bootstrap-options" \
   dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
   cluster-infrastructure="openais" \
   expected-quorum-votes="2" \
   stonith-enabled="false" \
   no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
   resource-stickiness="100"
```

Once we're happy with the changes, we can tell the cluster to start using them and use crm_mon to check everything is functioning.

cib commit drbd

quit

crm_mon

```
crm(drbd)# cib commit drbd
INFO: commited 'drbd' shadow CIB to the cluster
crm(drbd)# quit
bye
[root@test1 ~]# crm_mon
============
Last updated: Tue Sep  1 09:37:13 2009
Stack: openais
Current DC: test1 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
3 Resources configured.
============

Online: [ test1 test2 ]

ClusterIP (ocf::heartbeat:IPaddr):        Started test1
WebSite (ocf::heartbeat:apache):        Started test1
Master/Slave Set: WebDataClone
        Masters: [ test2 ]
        Slaves: [ test1 ]
```

Now that DRBD is functioning we can configure a Filesystem resource to use it. In addition to the filesystem's definition, we also need to tell the cluster where it can be located (only on the DRBD Primary) and when it is allowed to start (after the Primary was promoted).

Once again we'll use the shell's interactive mode

crm

cib new fs

configure primitive WebFS ocf:heartbeat:Filesystem \

params device="/dev/dev/mapper/vg_test1-drbd--demo" directory="/var/www/html" fstype="ext4"

configure colocation fs_on_drbd inf: WebFS WebDataClone:Master

configure order WebFS-after-WebData inf: WebDataClone:promote WebFS:start

```
[root@test1 ~]# crm
crm(live)# cib new fs
INFO: fs shadow CIB created
crm(fs)# configure primitive WebFS ocf:heartbeat:Filesystem params device="/dev/dev/mapper/vg_test1-drbd--
   demo" directory="/var/www/html" fstype="ext4"
crm(fs)# configure colocation fs_on_drbd inf: WebFS WebDataClone:Master
crm(fs)# configure order WebFS-after-WebData inf: WebDataClone:promote WebFS:start
```

We also need to tell the cluster that Apache needs to run on the same machine as the filesystem and that it must be active before Apache can start.

configure colocation WebSite-with-WebFS inf: WebSite WebFS

configure order WebSite-after-WebFS inf: WebFS WebSite

configure show

```
crm(fs)# configure colocation WebSite-with-WebFS inf: WebSite WebFS
crm(fs)# configure order WebSite-after-WebFS inf: WebFS WebSite
[root@test1 ~]# crm configure show
node test1
node test2
primitive WebData ocf:linbit:drbd \
    params drbd_resource="wwwdata" \
    op monitor interval="60s"
primitive WebFS ocf:heartbeat:Filesystem \
    params device="/dev/drbd/by-res/wwwdata" directory="/var/www/html" fstype="ext4"
primitive WebSite ocf:heartbeat:apache \
    params configfile="/etc/httpd/conf/httpd.conf" \
    op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
ms WebDataClone WebData \
    meta master-max="1" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
location prefer-test1 WebSite 50: test1
colocation WebSite-with-WebFS inf: WebSite WebFS
colocation fs_on_drbd inf: WebFS WebDataClone:Master
colocation website-with-ip inf: WebSite ClusterIP
order WebFS-after-WebData inf: WebDataClone:promote WebFS:start
order WebSite-after-WebFS inf: WebFS WebSite
order apache-after-ip inf: ClusterIP WebSite
property $id="cib-bootstrap-options" \
    dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
    cluster-infrastructure="openais" \
    expected-quorum-votes=”2” \
    stonith-enabled="false" \
    no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
    resource-stickiness=”100”
```

After reviewing the new configuration, we again upload it and watch the cluster put it into effect.

cib commit fs

quit

crm_mon

```
crm(fs)# cib commit fs
INFO: commited 'fs' shadow CIB to the cluster
crm(fs)# quit
bye
[root@test1 ~]# crm_mon
============
Last updated: Tue Sep  1 10:08:44 2009
Stack: openais
Current DC: test1 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
4 Resources configured.
============

Online: [ test1 test2 ]

ClusterIP (ocf::heartbeat:IPaddr):        Started test1
WebSite (ocf::heartbeat:apache):        Started test1
Master/Slave Set: WebDataClone
        Masters: [ test1 ]
        Slaves: [ test2 ]
WebFS   (ocf::heartbeat:Filesystem):    Started test1
```

Testing Migration

We could shut down the active node again, but another way to safely simulate recovery is to put the node into what is called “standby mode”. Nodes in this state tell the cluster that they are not allowed to run resources. Any resources found active there will be moved elsewhere. This feature can be particularly useful when updating the resources’ packages.

Put the local node into standby mode and observe the cluster move all the resources to the other node. Note also that the node’s status will change to indicate that it can no longer host resources.

crm node standby

crm_mon

```
[root@test1 ~]# crm node standby
[root@test1 ~]# crm_mon
============
Last updated: Tue Sep  1 10:09:57 2009
Stack: openais
Current DC: test1 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
4 Resources configured.
============

Node test1: standby
Online: [ test2 ]

ClusterIP (ocf::heartbeat:IPaddr):        Started test2
WebSite (ocf::heartbeat:apache):        Started test2
Master/Slave Set: WebDataClone
        Masters: [ test2 ]
        Stopped: [ WebData:1 ]
WebFS   (ocf::heartbeat:Filesystem):    Started test2
```

Once we've done everything we needed to on test1 (in this case nothing, we just wanted to see the resources move), we can allow the node to be a full cluster member again.

crm node online

crm_mon

```
[root@test1 ~]# crm node online
[root@test1 ~]# crm_mon
============
Last updated: Tue Sep  1 10:13:25 2009
Stack: openais
Current DC: test1 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
4 Resources configured.
============

Online: [ test1 test2 ]

ClusterIP (ocf::heartbeat:IPaddr):        Started test2
WebSite (ocf::heartbeat:apache):        Started test2
Master/Slave Set: WebDataClone
        Masters: [ test2 ]
        Slaves: [ test1 ]
WebFS   (ocf::heartbeat:Filesystem):    Started test2
```

Notice that our resource stickiness settings prevent the services from migrating back to test1.

Conversion to Active/Active

The primary requirement for an Active/Active cluster is that the data required for your services are available, simultaneously, on both machines. Pacemaker makes no requirement on how this is achieved, you could use a SAN if you had one available, however since DRBD supports multiple Primaries, we can also use that.

The only hitch is that we need to use a cluster-aware filesystem (and the one we used earlier with DRBD, ext4, is not one of those). GFS2 support is planed for later this year, however the best option currently is OCFS2. It comes with Fedora 11 and supports

- Variable Block sizes
- Flexible Allocations (extents, sparse, unwritten extents with the ability to punch holes)
- Journaling (ordered and writeback data journaling modes)
- Endian and Architecture Neutral (x86, x86_64, ia64 and ppc64)
- Support for Buffered, Direct, Asynchronous, Splice and Memory Mapped I/Os
- POSIX locks

Install a Cluster Filesystem - OCFS2

The first thing to do is install ocfs2-tools on each machine. Be sure to obtain the version from the same location as Pacemaker. This version is slightly newer and has improved Pacemaker support.

yum install -y ocfs2-tools

```
[root@test1 ~]# yum install -y ocfs2-tools
Loaded plugins: refresh-packagekit, versionlock
Reading version lock configuration
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package ocfs2-tools.x86_64 0:1.4.1-41.1 set to be updated
--> Processing Dependency: libdlm for package: ocfs2-tools-1.4.1-41.1.x86_64
--> Processing Dependency: libdlmcontrol.so.3()(64bit) for package: ocfs2-tools-1.4.1-41.1.x86_64
--> Running transaction check
---> Package libdlm.x86_64 0:2.99.08-15.1 set to be updated
---> Package libdlm2.x86_64 0:2.99.08-15.1 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package          Arch        Version            Repository                 Size
================================================================================
Installing:
 ocfs2-tools      x86_64      1.4.1-41.1         server_ha-clustering      832 k
Installing for dependencies:
```

```
 libdlm           x86_64      2.99.08-15.1       server_ha-clustering      107 k
 libdlm2          x86_64      2.99.08-15.1       server_ha-clustering       55 k

Transaction Summary
================================================================================
Install      3 Package(s)
Update       0 Package(s)
Remove       0 Package(s)

Total download size: 993 k
Downloading Packages:
(1/3): libdlm-2.99.08-15.1.x86_64.rpm                     | 107 kB     00:00
(2/3): libdlm2-2.99.08-15.1.x86_64.rpm                    |  55 kB     00:00
(3/3): ocfs2-tools-1.4.1-41.1.x86_64.rpm                  | 832 kB     00:00
--------------------------------------------------------------------------------
Total                                           39 kB/s | 993 kB     00:25
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing     : libdlm2-2.99.08-15.1.x86_64                              1/3
  Installing     : libdlm-2.99.08-15.1.x86_64                               2/3
  Installing     : ocfs2-tools-1.4.1-41.1.x86_64                            3/3

Installed:
  ocfs2-tools.x86_64 0:1.4.1-41.1

Dependency Installed:
  libdlm.x86_64 0:2.99.08-15.1           libdlm2.x86_64 0:2.99.08-15.1

Complete!
```

Setup Pacemaker-OCFS2 Integration

OCFS2 needs two services to be running, the first is the user-space interface to the kernel's distributed lock manager (DLM). The DLM is used to co-ordinate which node(s) can access a given file (and when) and integrates with Pacemaker to obtain node membership[16] information and fencing capabilities.

The second service is OCFS2's own control daemon which also integrates with Pacemaker to obtain node membership data.

Add the DLM service

The DLM control daemon needs to run on all active cluster nodes, so we will use the shells interactive mode to create a cloned resource.

crm

cib new stack-glue

configure primitive dlm ocf:pacemaker:controld op monitor interval=120s

configure clone dlm-clone dlm meta interleave=true

configure show

```
[root@test1 ~]# crm
crm(live)# cib new stack-glue
INFO: stack-glue shadow CIB created
crm(stack-glue)# configure primitive dlm ocf:pacemaker:controld op monitor interval=120s
crm(stack-glue)# configure clone dlm-clone dlm meta interleave=true
crm(stack-glue)# configure show xml
crm(stack-glue)# configure show
node test1
node test2
primitive WebData ocf:linbit:drbd \
    params drbd_resource="wwwdata" \
    op monitor interval="60s"
primitive WebFS ocf:heartbeat:Filesystem \
    params device="/dev/drbd/by-res/wwwdata" directory="/var/www/html" fstype="ext4"
primitive WebSite ocf:heartbeat:apache \
    params configfile="/etc/httpd/conf/httpd.conf" \
    op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
primitive dlm ocf:pacemaker:controld \
    op monitor interval="120s"
ms WebDataClone WebData \
    meta master-max="1" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
clone dlm-clone dlm \
    meta interleave="true"
location prefer-test1 WebSite 50: test1
colocation WebSite-with-WebFS inf: WebSite WebFS
colocation fs_on_drbd inf: WebFS WebDataClone:Master
colocation website-with-ip inf: WebSite ClusterIP
order WebFS-after-WebData inf: WebDataClone:promote WebFS:start
order WebSite-after-WebFS inf: WebFS WebSite
order apache-after-ip inf: ClusterIP WebSite
property $id="cib-bootstrap-options" \
    dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
```

[16] The list of nodes the cluster considers to be available

```
    cluster-infrastructure="openais" \
    expected-quorum-votes="2" \
    stonith-enabled="false" \
    no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
    resource-stickiness="100"
```

Review the configuration before uploading it to the cluster, quitting the shell and watching the cluster's response

cib commit stack-glue

quit

crm_mon

```
crm(stack-glue)# cib commit stack-glue
INFO: commited 'stack-glue' shadow CIB to the cluster
crm(stack-glue)# quit
bye
[root@test1 ~]# crm_mon
============
Last updated: Thu Sep  3 20:49:54 2009
Stack: openais
Current DC: test2 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
5 Resources configured.
============

Online: [ test1 test2 ]

WebSite (ocf::heartbeat:apache):        Started test2
Master/Slave Set: WebDataClone
        Masters: [ test1 ]
        Slaves: [ test2 ]
ClusterIP (ocf::heartbeat:IPaddr):        Started test2
Clone Set: dlm-clone
        Started: [ test2 test1 ]
WebFS   (ocf::heartbeat:Filesystem):    Started test2
```

Add the O2CB service

Once the DLM is active, we can add the OCFS2 control daemon.

Before we do so, check to see that the o2cb service is not being automatically started when the machine boots. This service is only used when OCFS2 uses its internal cluster stack. Neither it, nor the cluster.conf it creates should be used when using OCFS2 with Pacemaker.

To see if a node will start o2cb automatically, run

chkconfig | grep o2cb

```
[root@test1 ~]# chkconfig | grep o2cb
[root@test1 ~]#
```

There should be no output.

There are currently a few minor issues with the o2cb script that need to be addressed before we use it. To make the script compatible with Fedora 11, run the following commands (there should be no output).

sed -i.gres s/start_daemon//g /usr/lib/ocf/resource.d/ocfs2/o2cb

sed -i.gres s/pidofproc/pidof/g /usr/lib/ocf/resource.d/ocfs2/o2cb

Now use the **crm** shell to create the o2cb cluster resource:

crm

cib new oracle-glue

configure primitive o2cb ocf:ocfs2:o2cb op monitor interval=120s

configure clone o2cb-clone o2cb meta interleave=true

```
[root@test1 ~]# crm
crm(live)# cib new oracle-glue
INFO: oracle-glue shadow CIB created
crm(oracle-glue)# configure primitive o2cb ocf:ocfs2:o2cb op monitor interval=120s
crm(oracle-glue)# configure clone o2cb-clone o2cb meta interleave=true
```

Now ensure Pacemaker only starts the o2cb service on nodes that also have a copy of the dlm service (created above) already running

configure colocation o2cb-with-dlm INFINITY: o2cb-clone dlm-clone

configure order start-o2cb-after-dlm mandatory: dlm-clone o2cb-clone

```
crm(oracle-glue)# configure colocation o2cb-with-dlm INFINITY: o2cb-clone dlm-clone
crm(oracle-glue)# configure order start-o2cb-after-dlm mandatory: dlm-clone o2cb-clone
```

Review the configuration before uploading it to the cluster, quitting the shell and watching the cluster's response

configure show

cib commit oracle-glue

quit

crm_mon

```
crm(oracle-glue)# configure show
node test1
node test2
primitive WebData ocf:linbit:drbd \
    params drbd_resource="wwwdata" \
    op monitor interval="60s"
primitive WebFS ocf:heartbeat:Filesystem \
    params device="/dev/drbd/by-res/wwwdata" directory="/var/www/html" fstype="ext4"
primitive WebSite ocf:heartbeat:apache \
    params configfile="/etc/httpd/conf/httpd.conf" \
    op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
primitive dlm ocf:pacemaker:controld \
    op monitor interval="120s"
primitive o2cb ocf:ocfs2:o2cb \
    op monitor interval="120s"
ms WebDataClone WebData \
    meta master-max="1" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
clone dlm-clone dlm \
    meta interleave="true"
clone o2cb-clone o2cb \
    meta interleave="true"
location prefer-test1 WebSite 50: test1
```

```
colocation WebSite-with-WebFS inf: WebSite WebFS
colocation fs_on_drbd inf: WebFS WebDataClone:Master
colocation o2cb-with-dlm inf: o2cb-clone dlm-clone
colocation website-with-ip inf: WebSite ClusterIP
order WebFS-after-WebData inf: WebDataClone:promote WebFS:start
order WebSite-after-WebFS inf: WebFS WebSite
order apache-after-ip inf: ClusterIP WebSite
order start-o2cb-after-dlm inf: dlm-clone o2cb-clone
property $id="cib-bootstrap-options" \
   dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
   cluster-infrastructure="openais" \
   expected-quorum-votes="2" \
   stonith-enabled="false" \
   no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
   resource-stickiness="100"
crm(oracle-glue)# cib commit oracle-glue
INFO: commited 'oracle-glue' shadow CIB to the cluster
crm(oracle-glue)# quit
bye
[root@test1 ~]# crm_mon
============
Last updated: Thu Sep  3 20:49:54 2009
Stack: openais
Current DC: test2 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
6 Resources configured.
============

Online: [ test1 test2 ]

WebSite (ocf::heartbeat:apache):        Started test2
Master/Slave Set: WebDataClone
        Masters: [ test1 ]
        Slaves: [ test2 ]
ClusterIP (ocf::heartbeat:IPaddr):        Started test2
Clone Set: dlm-clone
        Started: [ test2 test1 ]
Clone Set: o2cb-clone
        Started: [ test2 test1 ]
WebFS   (ocf::heartbeat:Filesystem):    Started test1
```

Create an OCFS2 Filesystem

Preparation

Before we do anything to the existing partition, we need to make sure it is unmounted. We do this by tell the cluster to stop the WebFS resource. This will ensure that other resources (in our case, Apache) using WebFS are not only stopped, but stopped in the correct order.

crm_resource --resource WebFS --set-parameter target-role --meta --parameter-value Stopped

crm_mon

```
[root@test1 ~]# crm_resource --resource WebFS --set-parameter target-role --meta --parameter-value Stopped
[root@test1 ~]# crm_mon
============
Last updated: Thu Sep  3 15:18:06 2009
Stack: openais
Current DC: test1 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
6 Resources configured.
============

Online: [ test1 test2 ]

Master/Slave Set: WebDataClone
   Masters: [ test1 ]
   Slaves: [ test2 ]
ClusterIP (ocf::heartbeat:IPaddr):   Started test1
Clone Set: dlm-clone
        Started: [ test2 test1 ]
Clone Set: o2cb-clone
        Started: [ test2 test1 ]
```

Note that Apache and WebFS have both been stopped.

Create and Populate an OCFS2 Partition

Now that the cluster stack and integration pieces are running smoothly, we can create an OCFS2 partition.

Important: This will erase all previous content stored on the DRBD device. Ensure you have a copy of any important data.

The OCFS2 defaults are pretty good, so simply point mkfs to the DRBD partition we're going to use.

mkfs.ocfs2 /dev/drbd1

```
[root@test1 ~]# mkfs.ocfs2 /dev/drbd1
mkfs.ocfs2 1.4.2
Cluster stack: pcmk
Cluster name: pacemaker
NOTE: Selecting extended slot map for userspace cluster stack
Filesystem label=
Block size=1024 (bits=10)
Cluster size=4096 (bits=12)
Volume size=12541952 (3062 clusters) (12248 blocks)
1 cluster groups (tail covers 3062 clusters, rest cover 3062 clusters)
Journal size=4194304
Initial number of node slots: 2
Creating bitmaps: done
Initializing superblock: done
Writing system files: done
Writing superblock: done
Writing backup superblock: 0 block(s)
Formatting Journals: done
Formatting slot map: done
Writing lost+found: done
Formatting quota files: done
mkfs.ocfs2 successful

[root@test1 ~]#
```

Then (re)populate the new filesystem with data (web pages). For now we'll create another variation on our home page.

mount /dev/drbd1 /mnt/

cat <<-END >/mnt/index.html

<html>

<body>My Test Site - OCFS2</body>

</html>

END

umount /dev/drbd1

drbdadm verify wwwdata

```
[root@test1 ~]# cat <<-END >/mnt/index.html
> <html>
> <body>My Test Site - drbd</body>
> </html>
> END
[root@test1 ~]#
```

Reconfigure the Cluster for OCFS2

crm

cib new ocfs2

configure delete WebFS

configure primitive WebFS ocf:heartbeat:Filesystem \

params device="/dev/drbd/by-res/wwwdata" directory="/var/www/html" fstype="ocfs2"

```
[root@test1 ~]# crm
crm(live)# cib new ocfs2
INFO: ocfs2 shadow CIB created
crm(ocfs2)# configure delete WebFS
crm(ocfs2)# configure primitive WebFS ocf:heartbeat:Filesystem params device="/dev/drbd/by-res/wwwdata"
    directory="/var/www/html" fstype="ocfs2"
```

Now that we've recreated the resource, we also need to recreate all the constraints that used it. This is because the shell will automatically remove any constraints that referenced WebFS.

configure colocation WebSite-with-WebFS inf: WebSite WebFS

configure colocation fs_on_drbd inf: WebFS WebDataClone:Master

configure order WebFS-after-WebData inf: WebDataClone:promote WebFS:start

configure order WebSite-after-WebFS inf: WebFS WebSite

configure colocation WebFS-with-o2cb INFINITY: WebFS o2cb-clone

configure order start-WebFS-after-o2cb mandatory: o2cb-clone WebFS

configure show

```
crm(ocfs2)# configure colocation WebSite-with-WebFS inf: WebSite WebFS
crm(ocfs2)# configure colocation fs_on_drbd inf: WebFS WebDataClone:Master
crm(ocfs2)# configure order WebFS-after-WebData inf: WebDataClone:promote WebFS:start
crm(ocfs2)# configure order WebSite-after-WebFS inf: WebFS WebSite
crm(ocfs2)# configure colocation WebFS-with-o2cb INFINITY: WebFS o2cb-clone
crm(ocfs2)# configure order start-WebFS-after-o2cb mandatory: o2cb-clone WebFS
crm(ocfs2)# configure show
node test1
node test2
primitive WebData ocf:linbit:drbd \
    params drbd_resource="wwwdata" \
    op monitor interval="60s"
primitive WebFS ocf:heartbeat:Filesystem \
    params device="/dev/drbd/by-res/wwwdata" directory="/var/www/html" fstype="ocfs2"
primitive WebSite ocf:heartbeat:apache \
    params configfile="/etc/httpd/conf/httpd.conf" \
    op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
primitive dlm ocf:pacemaker:controld \
    op monitor interval="120s"
primitive o2cb ocf:ocfs2:o2cb \
    op monitor interval="120s"
ms WebDataClone WebData \
    meta master-max="1" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
```

```
clone dlm-clone dlm \
   meta interleave="true"
clone o2cb-clone o2cb \
   meta interleave="true"
colocation WebFS-with-o2cb inf: WebFS o2cb-clone
colocation WebSite-with-WebFS inf: WebSite WebFS
colocation fs_on_drbd inf: WebFS WebDataClone:Master
colocation o2cb-with-dlm inf: o2cb-clone dlm-clone
colocation website-with-ip inf: WebSite ClusterIP
order WebFS-after-WebData inf: WebDataClone:promote WebFS:start
order WebSite-after-WebFS inf: WebFS WebSite
order apache-after-ip inf: ClusterIP WebSite
order start-WebFS-after-o2cb inf: o2cb-clone WebFS
order start-o2cb-after-dlm inf: dlm-clone o2cb-clone
property $id="cib-bootstrap-options" \
   dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
   cluster-infrastructure="openais" \
   expected-quorum-votes=”2” \
   stonith-enabled="false" \
   no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
   resource-stickiness=”100”
```

Review the configuration before uploading it to the cluster, quitting the shell and watching the cluster's response

cib commit ocfs2

quit

crm_mon

```
crm(ocfs2)# cib commit ocfs2
INFO: commited 'ocfs2' shadow CIB to the cluster
crm(ocfs2)# quit
bye
[root@test1 ~]# crm_mon
============
Last updated: Thu Sep  3 20:49:54 2009
Stack: openais
Current DC: test2 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
6 Resources configured.
============

Online: [ test1 test2 ]

WebSite (ocf::heartbeat:apache):        Started test2
Master/Slave Set: WebDataClone
        Masters: [ test1 ]
        Slaves: [ test2 ]
ClusterIP (ocf::heartbeat:IPaddr):        Started test2
Clone Set: dlm-clone
        Started: [ test2 test1 ]
Clone Set: o2cb-clone
        Started: [ test2 test1 ]
WebFS   (ocf::heartbeat:Filesystem):    Started test1
```

Reconfigure Pacemaker for Active/Active

Almost everything is in place. Recent versions of DRBD are capable of operating in Primary/Primary mode and the filesystem we're using is cluster aware. All we need to do now is reconfigure the cluster to take advantage of this.

This will involve a number of changes, so we'll again use interactive mode.

crm

cib new active

There's no point making the services active on both locations if we can't reach them, so lets first clone the IP address. Cloned IPaddr2 resources use an *iptables* rule to ensure that each request only processed by one of the two clone instances. The additional *meta* options tell the cluster how many instances of the clone we want (one "request bucket" for each node) and that if all other nodes fail, then the remaining node should hold *all of them*. Otherwise the requests would be simply discarded.

configure clone WebIP ClusterIP \

meta globally-unique="true" clone-max="2" clone-node-max="2"

configure show

```
[root@test1 ~]# crm
crm(live)# cib new active
INFO: active shadow CIB created
crm(active)# configure clone WebIP ClusterIP  \
   meta globally-unique="true" clone-max="2" clone-node-max="2"
crm(active)# configure show
node test1
node test2
primitive WebData ocf:linbit:drbd \
   params drbd_resource="wwwdata" \
   op monitor interval="60s"
primitive WebFS ocf:heartbeat:Filesystem \
   params device="/dev/drbd/by-res/wwwdata" directory="/var/www/html" fstype="ocfs2"
primitive WebSite ocf:heartbeat:apache \
   params configfile="/etc/httpd/conf/httpd.conf" \
   op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
   params ip="192.168.9.101" cidr_netmask="32" \
   op monitor interval="30s"
primitive dlm ocf:pacemaker:controld \
   op monitor interval="120s"
primitive o2cb ocf:ocfs2:o2cb \
   op monitor interval="120s"
ms WebDataClone WebData \
   meta master-max="1" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
clone WebIP ClusterIP  \
   meta globally-unique="true" clone-max="2" clone-node-max="2"
clone dlm-clone dlm \
   meta interleave="true"
clone o2cb-clone o2cb \
   meta interleave="true"
colocation WebFS-with-o2cb inf: WebFS o2cb-clone
colocation WebSite-with-WebFS inf: WebSite WebFS
colocation fs_on_drbd inf: WebFS WebDataClone:Master
```

```
colocation o2cb-with-dlm inf: o2cb-clone dlm-clone
colocation website-with-ip inf: WebSite WebIP
order WebFS-after-WebData inf: WebDataClone:promote WebFS:start
order WebSite-after-WebFS inf: WebFS WebSite
order apache-after-ip inf: WebIP WebSite
order start-WebFS-after-o2cb inf: o2cb-clone WebFS
order start-o2cb-after-dlm inf: dlm-clone o2cb-clone
property $id="cib-bootstrap-options" \
   dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
   cluster-infrastructure="openais" \
   expected-quorum-votes="2" \
   stonith-enabled="false" \
   no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
   resource-stickiness="100"
```

Notice how any constraints that referenced *ClusterIP* have been updated to use *WebIP* instead. This is an additional benefit of using the **crm** shell.

Next we need to convert the filesystem and Apache resources into clones. Again, the shell will automatically update any relevant constraints.

```
configure clone WebFSClone WebFS
configure clone WebSiteClone WebSite
```

The last step is to tell the cluster that it is now allowed to promote both instances to be Primary (aka. Master).

```
configure edit WebDataClone
```

Change *master-max* to *2*

```
configure show
```

```
crm(active)# configure clone WebFSClone WebFS
crm(active)# configure clone WebSiteClone WebSite
crm(active)# configure edit WebDataClone
crm(active)# configure show
node test1
node test2
primitive WebData ocf:linbit:drbd \
   params drbd_resource="wwwdata" \
   op monitor interval="60s"
primitive WebFS ocf:heartbeat:Filesystem \
   params device="/dev/drbd/by-res/wwwdata" directory="/var/www/html" fstype="ocfs2"
primitive WebSite ocf:heartbeat:apache \
   params configfile="/etc/httpd/conf/httpd.conf" \
   op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
   params ip="192.168.9.101" cidr_netmask="32" \
   op monitor interval="30s"
primitive dlm ocf:pacemaker:controld \
   op monitor interval="120s"
primitive o2cb ocf:ocfs2:o2cb \
   op monitor interval="120s"
ms WebDataClone WebData \
   meta master-max="2" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
clone WebFSClone WebFS
clone WebIP ClusterIP  \
   meta globally-unique="true" clone-max="2" clone-node-max="2"
```

```
clone WebSiteClone WebSite
clone dlm-clone dlm \
   meta interleave="true"
clone o2cb-clone o2cb \
   meta interleave="true"
colocation WebFS-with-o2cb inf: WebFSClone o2cb-clone
colocation WebSite-with-WebFS inf: WebSiteClone WebFSClone
colocation fs_on_drbd inf: WebFSClone WebDataClone:Master
colocation o2cb-with-dlm inf: o2cb-clone dlm-clone
colocation website-with-ip inf: WebSiteClone WebIP
order WebFS-after-WebData inf: WebDataClone:promote WebFSClone:start
order WebSite-after-WebFS inf: WebFSClone WebSiteClone
order apache-after-ip inf: WebIP WebSiteClone
order start-WebFS-after-o2cb inf: o2cb-clone WebFSClone
order start-o2cb-after-dlm inf: dlm-clone o2cb-clone
property $id="cib-bootstrap-options" \
   dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
   cluster-infrastructure="openais" \
   expected-quorum-votes="2" \
   stonith-enabled="false" \
   no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
   resource-stickiness="100"
```

Review the configuration before uploading it to the cluster, quitting the shell and watching the cluster's response

cib commit active

quit

crm_mon

```
crm(active)# cib commit active
INFO: commited 'active' shadow CIB to the cluster
crm(active)# quit
bye
[root@test1 ~]# crm_mon
============
Last updated: Thu Sep  3 21:37:27 2009
Stack: openais
Current DC: test2 - partition with quorum
Version: 1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7
2 Nodes configured, 2 expected votes
6 Resources configured.
============

Online: [ test1 test2 ]

Master/Slave Set: WebDataClone
        Masters: [ test1 test2 ]
Clone Set: dlm-clone
        Started: [ test2 test1 ]
Clone Set: o2cb-clone
        Started: [ test2 test1 ]
Clone Set: WebIP
        Started: [ test1 test2 ]
Clone Set: WebFSClone
        Started: [ test1 test2 ]
```

```
    Clone Set: WebSiteClone
            Started: [ test1 test2 ]
```

Recovery

Configure STONITH

Why You Need STONITH

STONITH is an acronym for Shoot-The-Other-Node-In-The-Head and it protects your data from being corrupted by rouge nodes or concurrent access.

Just because a node is unresponsive, this doesn't mean it isn't accessing your data. The only way to be 100% sure that your data is safe, is to use STONITH so we can be certain that the node is truly offline, before allowing the data to be accessed from another node.

STONITH also has a role to play in the event that a clustered service cannot be stopped. In this case, the cluster uses STONITH to force the whole node offline, thereby making it safe to start the service elsewhere.

What STONITH Device Should You Use

It is crucial that the STONITH device can allow the cluster to differentiate between a node failure and a network one.

The biggest mistake people make in choosing a STONITH device is to use remote power switch (such as many onboard IMPI controllers) that shares power with the node it controls. In such cases, the cluster cannot be sure if the node is really offline, or active and suffering from a network fault.

Likewise, any device that relies on the machine being active (such as SSH-based "devices" used during testing) are inappropriate.

Configuring STONITH

1. Find the correct driver: stonith -L

2. Since every device is different, the parameters needed to configure it will vary. To find out the parameters required by the device: stonith -t {type} -n

   Hopefully the developers chose names that make sense, if not you can query for some additional information by finding an active cluster node and running:

   ```
   lrmadmin -M stonith {type} pacemaker
   ```

   The output should be XML formatted text containing additional parameter descriptions

3. Create a file called stonith.xml containing a primitive resource with a *class* of stonith, a *type* of {type} and a parameter for each of the values returned in step 2

4. Create a clone from the primitive resource if the device can shoot more than one node **and supports multiple simultaneous connections**.

5. Upload it into the CIB using cibadmin: cibadmin -C -o resources --xml-file stonith.xml

Example

Assuming we have an IBM BladeCenter containing our two nodes and the management interface is active on 192.168.9.31, then we would chose the external/ibmrsa driver in step 2 and obtain the following list of parameters

stonith -t external/ibmrsa -n

```
[root@test1 ~]# stonith -t external/ibmrsa -n
hostname  ipaddr  userid  passwd  type
```

Assuming we know the username and password for the management interface, we would create a STONITH resource with the shell

crm

cib new active

configure primitive rsa-fencing stonith::external/ibmrsa \

params hostname="test1 test2" ipaddr=192.168.9.31 userid=mgmt passwd=abc123 type=ibm \

op monitor interval="60s"

configure clone Fencing rsa-fencing

```
[root@test1 ~]# crm
crm(live)# cib new stonith
INFO: stonith shadow CIB created
crm(stonith)# configure primitive rsa-fencing stonith::external/ibmrsa \
   params hostname="test1 test2" ipaddr=192.168.9.31 userid=mgmt passwd=abc123 type=ibm \
   op monitor interval="60s"
crm(stonith)# configure clone Fencing rsa-fencing
```

And finally, since we disabled it earlier, we need to re-enable STONITH

configure property stonith-enabled="true"

configure show

```
crm(stonith)# configure property stonith-enabled="true"
crm(stonith)# configure show
node test1
node test2
primitive WebData ocf:linbit:drbd \
   params drbd_resource="wwwdata" \
   op monitor interval="60s"
primitive WebFS ocf:heartbeat:Filesystem \
   params device="/dev/drbd/by-res/wwwdata" directory="/var/www/html" fstype="ocfs2"
primitive WebSite ocf:heartbeat:apache \
   params configfile="/etc/httpd/conf/httpd.conf" \
   op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
   params ip="192.168.9.101" cidr_netmask="32" \
   op monitor interval="30s"
primitive dlm ocf:pacemaker:controld \
   op monitor interval="120s"
primitive o2cb ocf:ocfs2:o2cb \
   op monitor interval="120s"
primitive rsa-fencing stonith::external/ibmrsa \
   params hostname="test1 test2" ipaddr=192.168.9.31 userid=mgmt passwd=abc123 type=ibm \
   op monitor interval="60s"
```

```
ms WebDataClone WebData \
   meta master-max="2" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
clone Fencing rsa-fencing
clone WebFSClone WebFS
clone WebIP ClusterIP  \
   meta globally-unique=”true” clone-max=”2” clone-node-max=”2”
clone WebSiteClone WebSite
clone dlm-clone dlm \
   meta interleave="true"
clone o2cb-clone o2cb \
   meta interleave="true"
colocation WebFS-with-o2cb inf: WebFSClone o2cb-clone
colocation WebSite-with-WebFS inf: WebSiteClone WebFSClone
colocation fs_on_drbd inf: WebFSClone WebDataClone:Master
colocation o2cb-with-dlm inf: o2cb-clone dlm-clone
colocation website-with-ip inf: WebSiteClone WebIP
order WebFS-after-WebData inf: WebDataClone:promote WebFSClone:start
order WebSite-after-WebFS inf: WebFSClone WebSiteClone
order apache-after-ip inf: WebIP WebSiteClone
order start-WebFS-after-o2cb inf: o2cb-clone WebFSClone
order start-o2cb-after-dlm inf: dlm-clone o2cb-clone
property $id="cib-bootstrap-options" \
   dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
   cluster-infrastructure="openais" \
   expected-quorum-votes=”2” \
   stonith-enabled="true" \
   no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
   resource-stickiness=”100”
```

Configuration Recap

Final Cluster Configuration

crm configure show

```
[root@test1 ~]# crm configure show
node test1
node test2
primitive WebData ocf:linbit:drbd \
    params drbd_resource="wwwdata" \
    op monitor interval="60s"
primitive WebFS ocf:heartbeat:Filesystem \
    params device="/dev/drbd/by-res/wwwdata" directory="/var/www/html" fstype="ocfs2"
primitive WebSite ocf:heartbeat:apache \
    params configfile="/etc/httpd/conf/httpd.conf" \
    op monitor interval="1min"
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
primitive dlm ocf:pacemaker:controld \
    op monitor interval="120s"
primitive o2cb ocf:ocfs2:o2cb \
    op monitor interval="120s"
primitive rsa-fencing stonith::external/ibmrsa \
    params hostname="test1 test2" ipaddr=192.168.9.31 userid=mgmt passwd=abc123 type=ibm \
    op monitor interval="60s"
ms WebDataClone WebData \
    meta master-max="2" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
clone Fencing rsa-fencing
clone WebFSClone WebFS
clone WebIP ClusterIP  \
    meta globally-unique="true" clone-max="2" clone-node-max="2"
clone WebSiteClone WebSite
clone dlm-clone dlm \
    meta interleave="true"
clone o2cb-clone o2cb \
    meta interleave="true"
colocation WebFS-with-o2cb inf: WebFSClone o2cb-clone
colocation WebSite-with-WebFS inf: WebSiteClone WebFSClone
colocation fs_on_drbd inf: WebFSClone WebDataClone:Master
colocation o2cb-with-dlm inf: o2cb-clone dlm-clone
colocation website-with-ip inf: WebSiteClone WebIP
order WebFS-after-WebData inf: WebDataClone:promote WebFSClone:start
order WebSite-after-WebFS inf: WebFSClone WebSiteClone
order apache-after-ip inf: WebIP WebSiteClone
order start-WebFS-after-o2cb inf: o2cb-clone WebFSClone
order start-o2cb-after-dlm inf: dlm-clone o2cb-clone
property $id="cib-bootstrap-options" \
```

```
    dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
    cluster-infrastructure="openais" \
    expected-quorum-votes="2" \
    stonith-enabled="true" \
    no-quorum-policy="ignore"
rsc_defaults $id="rsc-options" \
    resource-stickiness="100"
```

Node List

The list of cluster nodes is automatically populated by the cluster.

```
node test1
node test2
```

Cluster Options

This is where the cluster automatically stores some information about the cluster

- dc-version - the version (including upstream source-code hash) of Pacemaker used on the DC
- cluster-infrastructure - the cluster infrastructure being used (heartbeat or openais)
- *expected-quorum-votes* - the maximum number of nodes expected to be part of the cluster

and where the admin can set options that control the way the cluster operates

- *stonith-enabled=true* - Make use of STONITH
- *no-quorum-policy=ignore* - Ignore loss of quorum and continue to host resources.

```
property $id="cib-bootstrap-options" \
    dc-version="1.0.5-462f1569a43740667daf7b0f6b521742e9eb8fa7" \
    cluster-infrastructure="openais" \
    expected-quorum-votes="2" \
    stonith-enabled="true" \
    no-quorum-policy="ignore"
```

Resources

Default Options

Here we configure cluster options that apply to every resource.

- *resource-stickiness* - Specify the aversion to moving resources to other machines

```
rsc_defaults $id="rsc-options" \
    resource-stickiness="100"
```

Fencing

```
primitive rsa-fencing stonith::external/ibmrsa \
    params hostname="test1 test2" ipaddr=192.168.9.31 userid=mgmt passwd=abc123 type=ibm \
    op monitor interval="60s"
clone Fencing rsa-fencing
```

Service Address

Users of the services provided by the cluster require an unchanging address with which to access it.

Additionally, we cloned the address so it will be active on both nodes. An *iptables* rule (created as part of the

resource agent) is used to ensure that each request only processed by one of the two clone instances. The additional *meta* options tell the cluster that we want two instances of the clone (one "request bucket" for each node) and that if one node fails, then the remaining node should hold *both*.

```
primitive ClusterIP ocf:heartbeat:IPaddr2 \
    params ip="192.168.9.101" cidr_netmask="32" \
    op monitor interval="30s"
clone WebIP ClusterIP  \
    meta globally-unique="true" clone-max="2" clone-node-max="2"
```

Distributed lock manager

Cluster filesystems like OCFS2 require a lock manager. This service starts the daemon that provides user-space applications (such as the OCFS2 daemon) with access to the in-kernel lock manager. Since we need it to be available on all nodes in the cluster, we have it cloned.

```
primitive dlm ocf:pacemaker:controld \
    op monitor interval="120s"
clone dlm-clone dlm \
    meta interleave="true"
```

Oracle control daemon

OCFS2 also needs a user-space/kernel bridge that runs on every node. So here we have another clone, however this time we must also specify that it can only run on machines that are also running the DLM (colocation constraint) and that it can only be started after the DLM is running (order constraint). Additionally, the o2cb clone should only care about the DLM instances it is paired with, so we need to set the interleave option.

```
primitive o2cb ocf:ocfs2:o2cb \
    op monitor interval="120s"
clone o2cb-clone o2cb \
    meta interleave="true"
colocation o2cb-with-dlm inf: o2cb-clone dlm-clone
order start-o2cb-after-dlm inf: dlm-clone o2cb-clone
```

DRBD - Shared Storage

Here we define the DRBD service and specify which DRBD resource (from drbd.conf) it should manage. We make it a master/slave resource and, in order to have an active/active setup, allow both instances to be promoted by specifying master-max=2. We also set the notify option so that the cluster will tell DRBD agent when it's peer changes state.

```
primitive WebData ocf:linbit:drbd \
    params drbd_resource="wwwdata" \
    op monitor interval="60s"
ms WebDataClone WebData \
    meta master-max="2" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
```

Cluster Filesystem

The cluster filesystem ensures that files are read and written correctly. We need to specify the block device (provided by DRBD), where we want it mounted and that we are using OCFS2. Again it is a clone because it is intended to be active on both nodes. The additional constraints ensure that it can only be started on nodes with active o2cb and drbd instances.

```
primitive WebFS ocf:heartbeat:Filesystem \
    params device="/dev/drbd/by-res/wwwdata" directory="/var/www/html" fstype="ocfs2"
clone WebFSClone WebFS
```

```
colocation WebFS-with-o2cb inf: WebFSClone o2cb-clone
colocation fs_on_drbd inf: WebFSClone WebDataClone:Master
order WebFS-after-WebData inf: WebDataClone:promote WebFSClone:start
order start-WebFS-after-o2cb inf: o2cb-clone WebFSClone
```

Apache

Lastly we have the actual service, Apache. We need only tell the cluster where to find it's main configuration file and restrict it to running on nodes that have the required filesystem mounted and the IP address active.

```
primitive WebSite ocf:heartbeat:apache \
    params configfile="/etc/httpd/conf/httpd.conf" \
    op monitor interval="1min"
clone WebSiteClone WebSite
colocation WebSite-with-WebFS inf: WebSiteClone WebFSClone
colocation website-with-ip inf: WebSiteClone WebIP
order apache-after-ip inf: WebIP WebSiteClone
order WebSite-after-WebFS inf: WebFSClone WebSiteClone
```

Add a Third Node

Appendix: Sample openais.conf

```
# Please read the openais.conf.5 manual page

aisexec {
   # Run as root - this is necessary to be able to manage resources with Pacemaker
   user:  root
   group: root
}

service {
   # Load the Pacemaker Cluster Resource Manager
   ver:       0
   name:      pacemaker
   use_mgmtd: yes
   use_logd:  yes
}

totem {
   version: 2

   # How long before declaring a token lost (ms)
   token:          5000

   # How many token retransmits before forming a new configuration
   token_retransmits_before_loss_const: 10

   # How long to wait for join messages in the membership protocol (ms)
   join:           1000

   # How long to wait for consensus to be achieved before starting a new round of membership configuration
   (ms)
   consensus:      2500

   # Turn off the virtual synchrony filter
   vsftype:        none

   # Number of messages that may be sent by one processor on receipt of the token
   max_messages:   20

   # Stagger sending the node join messages by 1..send_join ms
   send_join: 45

   # Limit generated nodeids to 31-bits (positive signed integers)
   clear_node_high_bit: yes
```

```
    # Disable encryption
    secauth:     off

    # How many threads to use for encryption/decryption
    threads:     0

    # Optionally assign a fixed node id (integer)
    # nodeid:         1234

    interface {
           ringnumber: 0

           # The following values need to be set based on your environment
bindnetaddr: 192.168.9.0
mcastaddr: 226.94.1.1
mcastport: 4000
    }
}

logging {
    debug: off
    fileline: off
    to_syslog: yes
    to_stderr: off
    syslog_facility: daemon
    timestamp: on
}

amf {
    mode: disabled
}
```

Appendix: Further Reading

Project Website

http://www.clusterlabs.org

Cluster Commands

A comprehensive guide to cluster commands has been written by Novell and can be found at:

http://www.novell.com/documentation/sles11/book_sleha/index.html?page=/documentation/sles11/book_sleha/data/book_sleha.html

OpenAIS

http://www.openais.org